A COMPUTATIONAL FRAMEWORK FOR NEONATAL BRAIN MRI STRUCTURE SEGMENTATION AND CLASSIFICATION

by

Rafael Ceschin

BS, University of Pittsburgh, 2010

MS, University of Pittsburgh, 2014

Submitted to the Graduate Faculty of

School of Medicine Department of Biomedical Informatics in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2017

UNIVERSITY OF PITTSBURGH

SCHOOL OF MEDICINE

This dissertation was presented

by

Rafael Ceschin

It was defended on

October 26, 2017

and approved by

Vanathi Gopalakrishnan, PhD

Ashok Panigrahy, MD

John Wu, PhD

Shandong Wu, PhD

Xinghua Lu, PhD

Dissertation Director: Vanathi Gopalakrishnan, PhD

ABSTRACT

A COMPUTATIONAL FRAMEWORK FOR NEONATAL BRAIN MRI STRUCTURE SEGMENTATION AND CLASSIFICATION

Rafael Ceschin, PhD

University of Pittsburgh, 2017

Deep Learning is increasingly being used in both supervised and unsupervised learning to derive complex patterns from data. However, the successful implementation of deep learning using medical imaging requires careful consideration for the quality and availability of data. Infants diagnosed with CHD are at a higher risk for neurodevelopmental impairment. Many of these deficits may be attenuated by early detection and intervention. However, we currently lack effective diagnostic tools for the reliable detection of these disorders at the neonatal period. We believe that the structural correlates of the cognitive deficits associated with developmental abnormalities can be measured within the first few months of life. Based on this assumption, we hypothesize that we can use an atlas registration based structural segmentation pipeline to sufficiently reduce the search space of neonatal structural brain MRI to viably implement convolutional neural networks for dysplasia classification. Secondly, we hypothesize that convolutional neural networks can successfully identify morphological biomarkers capable of detecting structurally abnormal brain substructures.

In this study, we develop a computational framework for the automated classification of dysplastic substructures from neonatal MRI. We validate our implementation on a dataset of neonates born with CHD, as this is a vulnerable population for structural dysmaturation. We chose the cerebellum as the initial test substructure because of its relatively simple structure and known vulnerability to structural dysplasia in infants born with CHD. We then apply the same method to the hippocampus, a more challenging substructure due to its complex

morphological properties. We attempt to overcome the limited availability of clinical data in neonatal populations by first extracting each brain substructure of interest and individually registering them into a standard space. This greatly reduces the search space required to learn the subtle abnormalities associated with a given pathology, making it feasible to implement a 3-D CNN as the classification algorithm. We achieved excellent classification accuracy in detecting dysplastic cerebelli, and demonstrate a viable computational framework for search space reduction using limited clinical datasets. All methods developed in this work are designed to be extensible, reproducible, and generalizable diagnostic tools for future neuroimaging problems.

Keywords: Deep Learning, Neonatal MRI, Congenital Heart Disease, Structural MRI .

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

I would not have accomplished this work without extensive help from family, friends, co-workers, and advisers. I would first like to thank Dr. Ashok Panigrahy, MD for his mentorship. Ashok trusted me with his research when I was still young and lost in my early career, and provided me with invaluable guidance that has unquestionably shaped who I am today. I would also like to thank Dr. Vanathi Gopalakrishnan, PhD for her incredible technical advice, life teachings, and patience throughout my career at DBMI. Vanathi taught me to be self-reflective about my research, and truly pushed me to make this dissertation the best work I could create. Thank you to my committee, Dr. John Wu, PhD, Dr. Shandong Wu, PhD, and Dr. Xinghua Lu, PhD for their support and guidance.

My time at DBMI has been productive, rewarding, and all around enjoyable. For that, I would like to thank Dr. Rebecca Jacobson for believing in me from the beginning. Thank you Dr. Roger Day, PhD for inspiring me personally and academically, and instilling a level of confidence I never had. Thank you to Dr. Jon Young, MD for the extensive discussions and ideas that helped shape this work. Additionally, I would like to thank Toni Porterfield for her endless support, advice, and encouragement. Thank you to the National Library of Medicine for supporting my training (5T15LM007059-27).

I would like to thank my work family at the Pediatric Imaging Research Center. Vince Lee for being a good friend and collaborator. Melanie Jenca for uncountable discussions and rants. Fern Wasco for being the most caring person in the world. Billy Reynolds and Alex Zahner for the hours of manual segmentation that made this entire work possible.

Finally, I dedicate this dissertation to my mother, Patricia Corby, who has always been my personal and profession role model, and my wife, Robin Ceschin, who has been my rock every step of the way.

GLOSSARY

Accuracy : A measure of how well a classification model predicts the desired classes. It is presented as a percentage, i.e. the fraction of cases it correctly predicts over the total number of cases.

ANTS : Advanced Normalization Tools is a state of the art image registration software. It performs both linear and non-linear transformations between a pair of images, outputting a symmetric transformation parameter that provides a mapping between the two images.

BET : Brain Extraction Tool is an algorithm that is part of the FSL software library that performs automated brain extraction of brain MRI.

Brain Segmentation/Parcellation : Subdividing a structural brain image into discrete anatomical (or functional) substructures by automated or manual methods.

Classification : The task of predicting the category that the input image (or datapoint in general) belongs to, given a finite set of classes.

CHD : Congenital Heart Disease.

CM : Ciliary motion is a scalar measure of how abnormal the cilia (microstructural cellular component) move, with 0 being normal ciliary motion and 4 being severely abnormal.

CNN : Convolutional Neural Networks are a specific category of neural networks that encode spatial information present in the input data in the form of feature maps.

CNS : Central nervous system.

CP : Choroid Plexus.

CRBNet : CeReBellum Network. 3-D Convolutional Neural Network for classification of dysplastic neonatal cerebelli.

CSF : Cerebral spinal fluid.

Deep Learning : The use of nested layers of neural networks. Typically any network that uses two or more layers can be considered a "deep" network.

DSC : Dice Similarity Coefficient is a measure of spatial similarity between two input images, ranging from 0 - 1, where 1 indicates identical images and 0 indication no overlapping voxels.

Dysmaturation : General term for any aberrant development of brain substructures.

For the purpose of this work, dysmaturation is further broken down into dysplasia and hypoplasia to differentiate volumetric and structural abnormalities. The mechanisms that lead to dysmaturation need further elucidation, but generally involve genetic and environmental factors such as perinatal injury.

Dysplasia : Form of dysmaturation that describes abnormal structural morphology (i.e. shape) of a given brain substructure, independent of volume.

DGM : Deep grey matter is a general term referring to several grey matter nuclei in the center of the brain, also referred to as basal ganglia.

Feature Map : A property of convolutional neural networks that encodes the spatial activation of a given set of weights and bias. In other words, a feature map records the presence of a given feature within a spatial representation of the input image.

FS : Freesurfer is the most widely used automated brain segmentation software within neuroimaging, however it is only optimized for fully developed brains.

FSL : FMRIB's Software Library is a software suite that contains a vast array of neuroimaging specific algorithms and pipelines.

Hyperparameter : One of several parameters that together comprise the architecture of a neural network, including the learning rate, number of layers, number of feature maps, activation function, etc...

Hypoplasia : Form of dysmaturation that describes an underdeveloped brain substructure based on volume.

IVH : Intraventricular hemorrhage.

LRF : Local receptive field is the size of the "scanning window" used by a CNN to search for a specific feature. This window is scanned across the entire image (convolution) and the presence of the feature at each location is encoded in a feature map.

Model : A trained neural network that encodes the necessary features of the input dataset to successfully predict the given classifier.

MRI : Magnetic Resonance Imaging

NeBSS : Neonatal Brain Structure Segmentation.

NN : Neural Networks are machine learning algorithms that apply a non-linear function to a linear combination of a given set of inputs. When nested, neural networks are capable

of approximating any highly complex non-linear functions, given enough neurons and layers.

Neuron : One unit of a neural network that applies a non-linear function to the linear combination of a given input.

PMA : Post-menstrual age, calculated in weeks, is post-birth age of a neonate plus their gestation age.

PVL : Periventricular leukomalacia is a complex pattern of white matter injury commonly associated with preterm birth.

Registration : Refers to either linear or non-linear transformation of one image onto the volumetric space of another. Also called normalization. Non-linear registration is also referred to as "warping".

RF : Radio frequency

Sequelae : Any pathology resulting from a prior illness, injury, or condition.

SGD : Stochastic gradient descent is a computational method of minimizing a cost function.

SNR : Signal-to-Noise Ratio

Substructure : Functional or anatomical components of the brain, such as cerebellum, hippocampus, etc...

Transfer Learning : Domain of machine learning that aims to leverage the parameters learned from training an algorithm on a (typically) large benchmark dataset by applying them to a new, independent dataset. This is typically achieved by coercing the input data into the same format as the benchmark dataset, and re-training the final classifier layer to classify the new dataset. This can be particularly powerful in the context of sparse, incomplete, or insufficient data.

Voxel : Most basic unit of a 3-D image (analogous to a pixel).

WMI : White matter injury

1.0 INTRODUCTION

Adapting theoretical applications of machine learning to generate practical solutions is one of the fundamental challenges in biomedical informatics. The directive is to apply cutting edge technology to solve real-world clinical problems with the goal of expanding our knowledge base and helping develop novel diagnostic and treatment strategies. However, algorithms that benchmark well on synthetic or neatly curated datasets often struggle to perform in real-world applications. Real data is noisy, limited, and highly varied. Therefore, inventive strategies are necessary to port these methods into meaningful applications. Generally, a divide and conquer strategy must be implemented in order to reduce broad, unwieldy tasks into manageable ones. In this thesis, we present a framework for implementing one of the most promising computational methods in the past decade on a challenging clinical dataset of neonatal structural MRI. We posit to use convolutional neural networks for both classification and inference to detect structural abnormalities in key brain substructures.

Structural MRI gives us the ability to parcellate a subject's brain into local substructures, giving us much finer granularity and specificity when diagnosing and treating localized structure-associated cognitive deficits. By extracting each brain structure of interest individually, we greatly reduce the search space required to learn the subtle abnormalities associated with a given pathology. Furthermore, by modeling our research question as a simpler image classification model, we can leverage the recent resurgence of neural networks as a powerful set of tools in image recognition and computer vision.

Deep neural networks, or deep learning, are a set of machine learning algorithms that use nested layers of linear combinations of the original input which allow for the approximation of highly complex non-linear functions [1]. This property can be used in both supervised learning for classification tasks and unsupervised learning to derive complex patterns from

the input data. Deep neural networks have recently gained traction across a variety of domains, but none more than in imaging and computer vision. Modern implementations, notably AlexNet [2], have been overwhelmingly successful in image classification challenges with upwards of 1000 different classification categories, far outperforming competing methods. Naturally, the application of deep neural networks to clinical inference, biomarker discovery, and automated diagnosis presents innumerable opportunities.

Medical imaging applications ranging from digital pathology stain normalization [3] to classifying skin lesion images [4] have employed variants of neural networks to solve classification problems. Within the domain of neuro-imaging, deep learning has largely been successful in intensity-based segmentation tasks. Kleesiek et al. [5] successfully implemented a 3-D CNN that outperforms the state-of-the-art algorithms in skull stripping, generalizing well to multi-modal inputs including contrast-enhanced images. Brosch et al. [6] used 3-D CNNs to segment white matter lesions in brain MRI from patients with multiple sclerosis. Across domains, deep learning has proven to be an invaluable tool, having the potential to not only achieve near-human levels in classification tasks, but also to provide unprecedented objective insight into the mechanisms that drive human intuition. Despite this gain in popularity, however, neural networks can still be prohibitively difficult to implement in a large subset of classification tasks.

1.1 THE PROBLEM

1.1.1 Neuroimaging Applications

The implementation of a neuroimaging data processing pipeline requires the development of robust and reproducible methods while minimizing user input and manipulation. However, standardizing highly variable clinical data originating from multiple sources is not a trivial task. Variations in MR scanner, clinical protocol, and pathology can greatly hinder performance. Not surprisingly, most existing processing methods are optimized using standardized control populations, with a very limited margin of error both in protocol and structural mor-

phology allowed. Efforts have been made to create standardized processing pipelines for a variety of classical MR processing tasks.[7] However, applying existing methods to neonatal populations eschews established data processing conventions. The underlying biology and anatomy is not sufficiently comparable, we do not have the luxury of large standardized datasets, and directly applying classification methods using the raw imaging output yields inadequate results.

Instead, we must develop highly customized pipelines to overcome these challenges, modifying existing tools to better suit the ground truth. Computationally, a pipeline architecture has the advantage of compartmentalizing each step into discrete processing nodes with their own input and output. This has two benefits. One, it permits the chaining of various independently developed neuroimaging packages, algorithms, and software without having to explicitly dictate how they interact with one another. Two, it enables the parallelization of processing streams, greatly improving computational performance and allows for the distributed processing of data across multiple computing nodes. We will use this to our advantage, and narrow our scope to two distinct problems: brain substructure segmentation followed by substructure morphology characterization.

1.1.2 Neonatal Structural Segmentation

Substructure segmentation of fully developed brains is a task that has been iteratively optimized over several decades. Acquisition protocols are largely standardized, and the inherent brain structure and tissue MR contrast is stable through adulthood. As a result, robust industry standards have been developed from sufficiently large training datasets. The same application in neonatal imaging, however, has yet to achieve any acceptable level of consensus, in both acquisition and computational approaches.

There are two primary factors that impede the use of adult optimized protocols in image acquisition of developing infants. First, the neonatal brain has a much higher water content at birth, with rapid decrease in water content in both grey and white matter as the brain develops. Second, the neonatal white matter is largely unmyellinated at birth. More importantly, both of these properties change very rapidly through the first year of development.

This leads to unique sequence development challenges, requiring age specific optimization at week-level intervals. The acquisition of neonatal MRI provides further difficulty, as smaller brains provide a lower signal to noise ratio (SNR) and neonates are more susceptible to motion in the loud MRI environment. Finally, the large morphological variability observed in clinically abnormal populations, especially pathology leading to ventriculomegaly and white matter lesions, require comparatively more flexible algorithms. Computationally, both the spatial and intensity priors used in the optimization of adult methods are generally inadequate. The change in tissue contrast is significant enough that ideally we would require a different set of priors for each gestational week until development reaches early-adulthood. Taken together, these factors make the acquisition of large neonatal datasets both labor and cost prohibitive. Therefore, alternative methods that leverage probabilistic atlases and expert-guided manual correction must be implemented.

1.1.3 Neural Networks

There are several challenges associated with deep neural networks. While computationally accessible, large datasets and deep architectures still require powerful hardware and long computation times to learn predictive models or classifiers. Furthermore, the accuracy of the classifier is highly dependent on the variance encoded within the training dataset. Intensity-based segmentation tasks have the advantage of dense training datasets, with relatively low complexity modeled by the feature sets. In contrast, structural morphology-based classification tasks, as presented in this study, suffer from sparse data and the need for highly complex features to model the structural complexity of brain substructures such as the hippocampus or cerebellum. As such, very large datasets are often necessary for more complex classification tasks. This is particularly difficult when implementing CNNs in medical imaging, as there can often be an insufficiently low incidence of abnormal cases in the training dataset. This can significantly slow down training, or impede it altogether, as the algorithm sees a disproportionate amount of normal cases through each iteration. Existing strategies to attenuate these effects will vary depending on the domain and application, but in general involve the reduction of the search space, removing noise from the input data, and enhancing

the signal through bootstrapping and pre-processing. Finally, while some efforts have been made in developing strategies for visualizing and understanding the features learned by the model [8], the intuitive interpretability of these features is not guaranteed.

1.1.4 Structural Dysmaturation in Neonates

The long term neurocognitive effects of perinatal complications have long been a challenge in child development. Neurocognitive abnormalities such as motor and executive function deficits, broad learning disabilities, poor mathematical performance, and ADHD are all examples of long term complications which may be attenuated by early detection and intervention [9]. We currently lack effective diagnostic tools for the reliable detection of these disorders at the neonatal period, even with known strong structure-function associations observed in key brain structures later in life [10]. Diagnosis of subtle structural abnormalities in neonatal imaging is prohibitively labor intensive, and requires a highly specialized expert to do so. Furthermore, existing clinical guidelines are qualitative in nature and lack sufficient specificity and reproducibility to be relied upon for the development of treatment strategies.

Term born neonates with congenital heart disease (CHD) are also more likely to present with mild ischemic injuries and periventricular leukomalacia (PVL) [11], mediated by an associated genetic component along with the environmental factors. The long term effects of these devastating neonatal injuries are multi-faceted, and can range from severe motor deficits such as cerebral palsy, to broad cognitive deficits including autism and ADHD [12]. Incidentally, the associated biomarkers of severe injuries are more easily identified in conventional imaging. In recent decades, however, improvements in palliative care and surgical interventions have greatly mitigated the severity of the initial injuries, as well as their associated poor outcomes, in these high risk neonatal populations. This has led to the advent of a new set of complications which are much more subtle to detect, and subsequently treat. We now observe subtle micro- and macrostructural, as well as aberrant connectivity variances that may go undetected at an age where early intervention could greatly mitigate the long term effects of these injuries. Furthermore, the sequellae may not present at an observable level until much later in the child's development, inhibiting our ability to preemptively

develop an effective treatment plan.

1.2 THE APPROACH

In this study, we develop a computational framework for the automated classification of dysplastic substructures from neonatal MRI. We validate our implementation on a dataset of neonates born with CHD, as this is a vulnerable population for structural dysmaturation. We chose the cerebellum as the initial test substructure because of its relatively simple structure and known vulnerability to structural dysplasia in infants born with CHD. We then apply the same method to the hippocampus, a more challenging substructure due to its complex morphological properties. We attempt to overcome the limited availability of clinical data in neonatal populations by first extracting each brain substructure of interest and individually registering them into a standard space. This greatly reduces the search space required to learn the subtle abnormalities associated with a given pathology, making it feasible to implement a 3-D CNN as the classification algorithm. Additionally, by enforcing spatial localization in the input data, the feature maps retain their spatial relation to the original structural morphology. The benefit of this approach is two-fold. First, once the model is derived from proper tuning of its parameters based on the training data, its implementation as a classification tool is straightforward, providing a feasible application of automated diagnostic systems in medical imaging. Second, the algorithm generates human-interpretable activation maps of the hidden layers learned by the network, thereby retaining the original 3-D structural relationships. This gives us a data-driven model of the features within the dataset that contribute to the final classification, providing further insight into the structural morphology associated with the classification criteria and underlying pathology. Figure 1 depicts a graphical representation of the framework.

This framework is divided into two components. First, the Neonatal Brain Structure Segmentation (NeBSS) pipeline is a semi-automated algorithm that takes as input neonatal structural MRI and generates 50 discrete brain substructures based on a neonate-specific brain atlas. Second, CRBNet is an implementation of a 3-D convolutional neural network

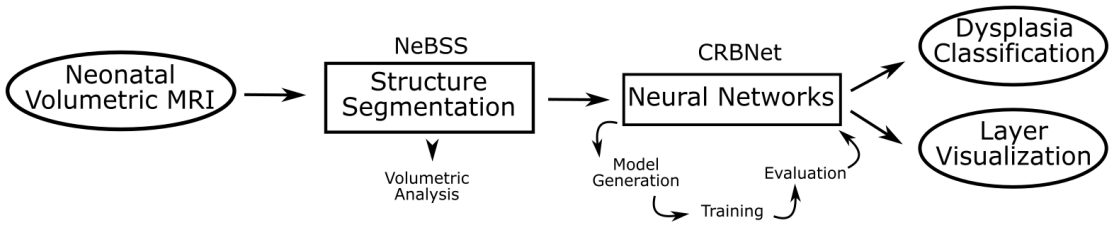

Figure 1: Overall design of the framework

that is able to identify structural malformation in individual brain substructures based on their morphological characteristics. This is a powerful framework that has the potential to improve on existing automated clinical diagnostic methods by heuristically modeling the strategy used by domain experts to diagnose structural abnormalities in individual brain substructures. It is provided as an open source, extensible, and flexible set of pipelines designed to be generalizable to image classification domains beyond neonatal imaging.

1.2.1 Thesis

The structural correlates of the cognitive deficits associated with developmental abnormalities can be measured within the first few months of life. Based on this assumption, we can use an atlas registration based structural segmentation pipeline to sufficiently reduce the search space of neonatal structural brain MRI to viably implement convolutional neural networks for dysplasia classification. Secondly, we hypothesize that convolutional neural networks can successfully identify morphological biomarkers capable of detecting structurally abnormal brain substructures.

This thesis can be stated using the following two strong claims:

1. NeBBS is a robust segmentation pipeline capable of segmenting brain substructures on clinically acquired neonatal populations.

2. CRBNet is a framework for training convolutional neural networks capable of achieving near-human level classification accuracy on real-world clinical data.

Additionally, the following weak claim is made:

1. The features learned can be used to inform and direct the development of more rigid clinical guidelines.

1.3 SIGNIFICANCE

This work is jointly relevant to both biomedicine and neuroinformatics, as it aims to implement state of the art machine learning methods into a highly specialized tool for diagnosing a burdensome and complex biological problem. Automated neonatal parcellation is a challenging computational task that is necessary for the effective diagnosis and treatment of structure-function associated neurodevelopmental deficits in several at-risk neonatal populations. Developing a parcellation pipeline that is robust to both image quality and acquisition variability is indispensable in clinical translational research, particularly in the setting of multi-site research studies and retrospectively acquired clinical data. NeBSS is designed to fill this need, with the primary objective of being flexible and robust when handling variations in input. Additionally, this work has the potential of developing more rigorous clinical guidelines for diagnosing structural abnormalities in neonates, by using the features extracted from the neural networks to further refine the current qualitative scoring method.

Moreover, we believe the successful implementation of a comprehensive framework for convolutional neural networks using 3-dimensional anatomical substructures to be a significant contribution to both translational medicine and informatics. To our knowledge, no neuro-imaging specific implementation of this framework exists in an easily accessible form to clinical researchers. Implementing a search space reduction algorithm to clinical data presents a viable method of utilizing an otherwise prohibitive machine learning method to real world, limited clinical data. Finally, all methods developed in this work are designed with the overarching goal of creating extensible, reproducible, and generalizable diagnostic tools for future neuroimaging problems.

1.4 THESIS OVERVIEW

Figure 2 shows a road map of the framework and where each specific topic is covered in this document.

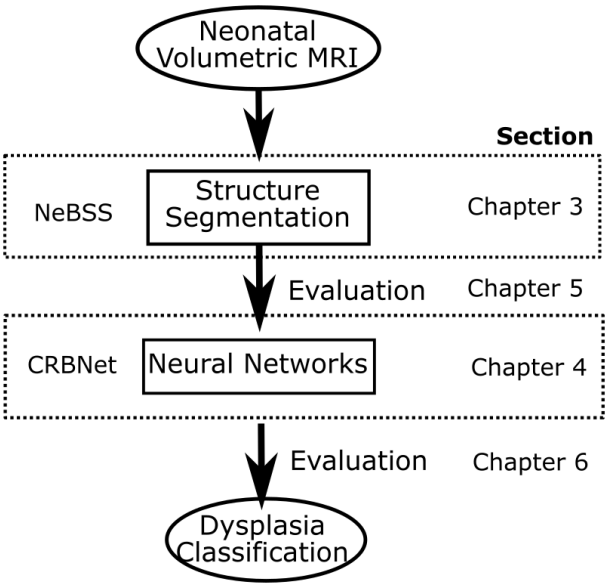

Figure 2: General road map of thesis

The second chapter will present a brief literature review of the biological problem and general machine learning background necessary for the development of this work. In chapter 3 I will describe in detail the methods and software implementation of NeBSS. This chapter covers the existing neuroimaging tools used by NeBSS, followed by the general architecture of the pipeline and current software implementation and workflow. Finally, this chapter presents the strategy used to evaluate the segmentation accuracy of the software. The results of this work are presented in chapter 4. I discuss the workflow and its reliability across users and repeated measures. Additionally, we evaluate its performance on data acquired across multiple magnetic field strengths, and provide a case study of its use on a clinical dataset. Similarly, chapter 5 covers in more depth the computational methods required to implement a working convolutional neural network. Here I will also describe the strategy used to design a successful architecture, including model creation, training strategies, and

evaluation methods. This chapter also presents the software implementation of 3-D CNNs, and presents a strategy for evaluating its performance. The evaluation results are presented in chapter 6.

2.0 BACKGROUND

2.1 STRUCTURAL DYSMATURATION IN NEONATES

The two primary injuries associated with structural dysmaturation observed in neonates can be classified into two broad types: hypoplasia and dysplasia [13]. Hypoplasia specifically refers to the underdevelopment of a given brain substructure. The substructure may retain its normal shape, but has significantly lower volume for the subject's age. The mechanisms that lead to hypoplasia can be due to stunted development, but also due to destructive mechanisms as a result of prior injury or genetic predisposition. Hypoplasia can be directly measured using volumetric approaches, and is generally straightforward to compare across populations. Dysplasia, on the other hand, describes a substructure of abnormal shape that may otherwise retain a normative volume. Dysplasia can be much more difficult to assess and quantify, requiring a trained expert in neuroanatomy with significant experience. The criteria for structural dysplasia varies across each brain substructure, and at the present lacks formal quantitative guidelines. Hypoplasia and dysplasia are not mutually exclusive, and it is imperative for this study to differentiate each diagnosis to better delineate the underlying injury mechanism and improve our detection of each pathology independently.

Here, we will primarily focus on CHD and preterm born neonates, due to our accessibility to a large cohort of each and the potential impact in these at risk populations. However, it is our hope that this work will be generalizable to broader studies beyond neonatal imaging.

2.1.1 Prematurity

Historically, infants born preterm have been shown to be at much higher risk of severe brain injuries, often presenting as diffuse white matter loss, periventricular leukomalacia (PVL), and associated intraventricular hemmorrhage (IVH)[14]. The mechanism of injury is not fully understood, but it is hypothesized to be partially caused by a combination of intrauterine and/or postnatal ischaemia and inflammation. Neonates born preterm have been shown to exhibit accelerated post-natal somatic growth when compared to term born neonates [15]. This phenomenon, commonly referred to as a catch-up growth period, places the neonate at an increased risk for developmental deficits, and tends to be more severe in very early preterm births. The long term effects of aberrant development during the catch up period bear strong association with accompanying severe growth restrictions, and observed abnormalities can include decrease in brain volume, increase in pituitary height, and thinning of the corpus callosum. Studies investigating the effects of symmetrical growth restriction (both head circumference and somatic growth restricted) compared to asymmetrical growth (preservation of head growth in a low weight neonate) in preterm born neonates show that both populations are at higher risk for developmental delay independent of head circumference at birth, when compared to non-growth restricted preterms. However, symmetrical growth restricted preterms showed an increase catch up growth in head circumference [16]. This suggests that while restricted growth can be a predictor of short term developmental deficits, it does not hold the specificity required for accurate prediction of long term impairment and development of appropriate targeted therapies. Here, I will highlight two substructures in the developing preterm born neonate that are known to be at a significant risk of maldevelopment: the cerebellum and hippocampus.

The cerebellum undergoes rapid development late in the gestation period, at which time a disrupted catch up growth period may lead to irreparable long term damage. This places the cerebellum at particularly high risk for developmental impairment in preterm born neonates. The primary function of the cerebellum has classically been associated with motor learning, tone, coordination, and language. However, early injury to the cerebellum shows evidence of mediating impairment in known afferent contralateral remote cortical regions [17], leading to

reduced growth in cerebral cortex by age 3 [18]. The most prevalent pathology affecting the cerebellum in premature born infants are cerebellar atrophy and hypoplasia [19], and have implications on domain specific functional deficits [17]. There are many factors thought to contribute to cerebellar hypoplasia. Exposure to glucocorticoids, common in treatment of hypotension and to accelerate lung maturation, is known to inhibit sonic hedgehog pathways (SHH) critical to cerebellar development. Additionally, direct exposure to blood products and hemosiderin from cerebellar hemorrhage can have a direct effect on cerebellar growth. Finally, cerebral injury such as intra-ventricular hemorrhage (IVH) and more severe white matter injury (WMI) can lead to downstream disruption of cerebellar development [20].

The preterm hippocampus, in contrast, is less susceptible to hypoplasia when compared to term controls. It does, however show evidence of dysplasia measured later in life, and is strongly correlated with very preterm infants [21]. The function of the hippocampus primarily correlates with memory tasks. However, dysplasia in children born preterm seems to only act as a weak predictor of memory outcome [22]. The mechanism of injury to the preterm hippocampus is not as well understood as the cerebellum, but recent work shows a correlation between Midazolam dose, a common sedative given to neonates, and hippocampal growth and neurodevelopmental outcome [23].

2.1.2 Congenital Heart Disease

Infants diagnosed with CHD are at a higher risk for neurodevelopment impairment, with particular predilection for motor deficits [24, 25, 26]. Term born neonates with CHD are more likely to present with mild ischemic injuries and PVL, similar to preterm born neonates [11]. Additionally, studies have demonstrated globally decreased brain volumes in neonates with CHD [27]. While post-natal surgical intervention has been shown to increase this risk, it does not account for all observed deficits, suggesting a more complex perinatal mechanism of injury [28, 29].

Pre-op regional volumetric growth differences have been observed in neonates, mainly in the frontal lobes and brainstem. However, this impaired growth seems to be independent of focal white matter injury, indicating additional genetic factors are at play [30]. This study,

however, used very crude methods of brain metrics, and more granularity is needed. To further suggest genetic factors as the driving force behind aberrant development in CHD neonates, it has been shown that, unlike in preterm born neonates, asymmetrical growth (head circumference continues to grow while somatic growth is stunted) is not a brain sparing adaption in CHD infants who underwent single ventricle reconstruction (SVR) [31]. Longitudinal studies have shown that these volumetric deficits tend to subside, and term born neonates with CHD seem to follow normative growth rates through infancy, independent of surgical procedure. One exception to this normalization of growth, however, is the cerebellum [32], which showed decreased volume even at a later age [33].

More recent work seeking to identify the genetic correlates between CHD and neurodevelopmental deficits has brought forward a compelling theory suggesting the maldevelopment of primary and motile cilia as a potential mediator of poor outcomes. Abnormal ciliary motion (CM) has previously been implicated as a primary cause of a wide range of syndromes affecting the central nervous system, including fetal hydrolethalus and acrocallosal syndrome [34]. More indirectly, it has been established that the Sonic hedgehog (SHH) pathway, important in neurogenesis and the development of key brain substructures including the cerebellum, requires specific protein trafficking pathways found in primary cilium in order to properly function [35]. To further support this theory, respiratory ciliary dysfunction has been observed in CHD patients with heterotaxy [36], as well as in mouse models indicating a central role for cilia that extends beyond the CNS [37]. The sequelae of CM abnormality in mouse models has been observed as neonatal hydrocephalus [38].

Our work shows a strong correlation between abnormal CM and maldevelopment of key brain substructures [39], consistent with the genetic mouse models and known ciliary dysfunctions. Figure 3 shows selected examples covering the spectrum of brain abnormalities observed in infants diagnosed with CHD and abnormal CM. While the severe abnormalities are much more easily detectable, the mild to moderate impairment can be challenging to diagnose.

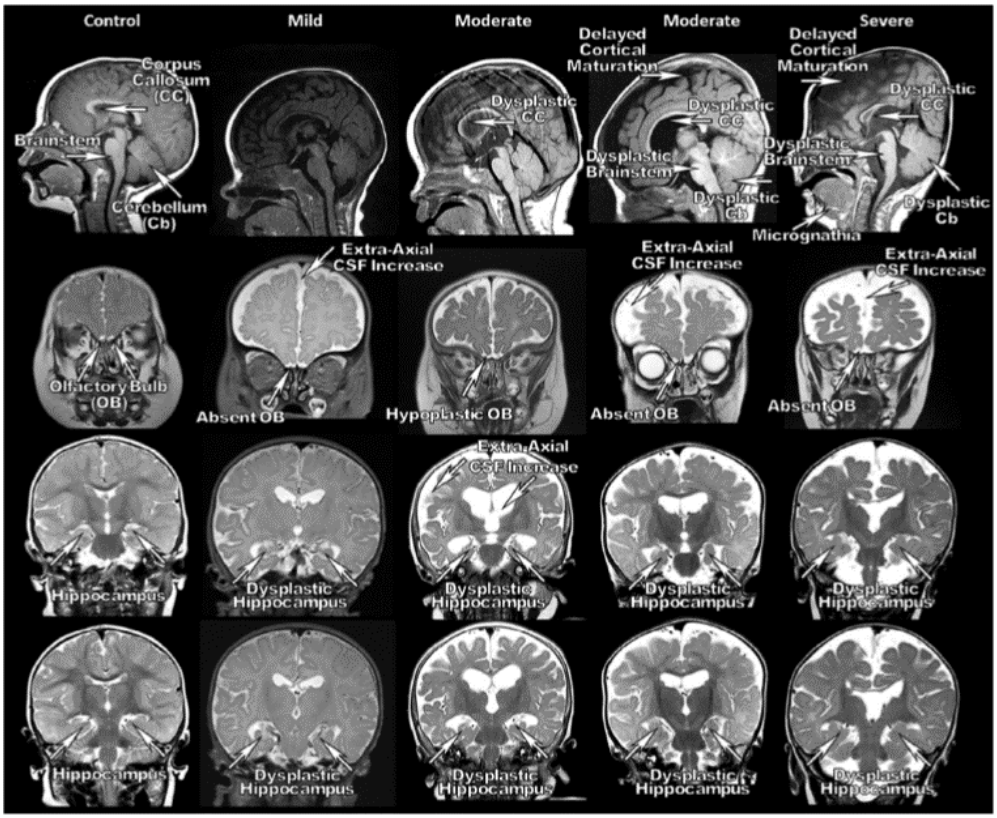

Figure 3: Brain abnormalities in infants with CHD and abnormal CM (from [39])

2.1.3 Brain Dysmaturation Criteria

No previous work has attempted to accurately quantitate the degree of brain dysmaturation in neonates born with CHD. In order to more precisely gauge the degree of injury in this population, we have developed a composite score [39] of total brain dysmaturation, based on a set of substructures known to be abnormal in genetic mouse models with CHD. The dysmaturation score is comprised of an aggregate sum of the following qualitative observations:

- Cerebellar Hypoplasia
- Cerebellar Dysplasia
- Cerebellar Vermis Hypoplasia
- Cerebellar Vermis Dysplasia
- Supratentorial Extra-Axial Fluid
- Right/Left Olfactory Bulb
- Right/Left Olfactory Sulci
- Right/Left Hippocampal Hypoplasia
- Right/Left Hippocampal Dysplasia
- Corpus Callosum Malformation
- Choroid Plexus Abnormality
- Brainstem Dysplasia

All observations are measured using conventional MRI, with the exception of Choroid Plexus (CP) abnormality, which is measured using conventional cranial ultrasound.

Volumetric measures are easier to compute (determine hypoplasia) and are calculated directly from the segmentation pipeline. Morphology measures, which determine dysplasia, are harder to reliably quantify, as we need rigorous guidelines for determining abnormal "shape." These measures are very labor intensive, and rely on trained anatomy and pathology experts with significant previous experience.

2.2 STRUCTURAL MRI IN NEONATES

2.2.1 Image Acquisition

Conventional ^1H MRI takes advantage of the physical properties of protons (hydrogen nuclei in polar covalent bonds in fat or water molecules) when in the presence of a strong magnetic field. The magnetic field strength in a typical clinical MRI ranges from 1.5 - 3.0 Tesla, which is around thirty thousand times the strength of earth's magnetic field. In this environment, molecules with an odd number of nucleons such as ^1H, abundantly present in water and fat, will align their axis of precession with that of the local magnetic field. When we apply a radio pulse precisely at the resonating frequency of the molecule's precession, the molecules absorb a small amount of energy and the axis of precession flips vertically with respect to the orientation of the magnetic field, and will all initially synchronize their precession. MRI acquires an image by measuring the localized energy released when the orientation of these hydrogen nuclei snaps back into their original position; this is called relaxation. The contrast between tissue types observed across the various types of imaging is a result of the precise timing at which we measure this release of energy. This is called the Time to Echo (TE), and it is conventionally optimized to maximize the signal measured from either water or fat molecules (in structural imaging). Thus the image intensity at each voxel is the average signal measured at that specific TE, and is proportional to the composition of the underlying tissue [40].

When we talk about structural imaging, we generally refer to either T1 or T2 weighted images. T1 imaging directly measures the longitudinal relaxation time of the hydrogen nuclei, i.e. how quickly the nuclei snap back to their relaxed position. T2 imaging measures what is referred to as the transverse relaxation time, and is indicative of how quickly a proton's precession goes out of phase with respect to their neighboring nuclei. Defined tissue contrast in MRI is made possible because the underlying structure of molecules directly affects both their T1 and T2 relaxation times. In this context, fat is a more structured tissue type compared to water, as a proton lattice exists as a result of its rigid carbon chain, compared to more randomly distributed water molecules in fluid. In simplified terms, a proton lattice

allows for a more rapid dispersion of the energy absorbed from the RF pulse, resulting in faster T1 and T2 relaxation times. It is important to emphasize that in practice we acquire T1 and T2 *weighted* images, where we measure an optimal combination of the tissue's T1 and T2 properties by varying both the interval at which we apply the RF pulse (called the repetition time - TR), and the interval at which we measure the signal (TE).

Heuristically, T1 weighted imaging will have much shorter TR and TE, and tissue with higher fat content will appear brighter in the image. With T2 weighted imaging, the TR and TE will be longer, so the signal from higher fat content decays before we measure the release of energy, and we are left with the signal from tissue with slower decaying relaxation, i.e. higher water content. This property is the driving force behind neuroimaging. The three main tissue types visible in brain MRI are white matter, grey matter, and cerebral spinal fluid (CSF). In fully developed brains, the axon portion of the neurons, collectively known as white matter, is wrapped in myelin, a fatty insulating sheathe. This gives the axons a much shorter relaxation time, which makes white matter highly visible in T1 weighted imaging. Conversely, CSF is primarily water, and therefore has a longer relaxation time, showing up brightly on T2 weighted images. Grey matter conveniently falls in between white matter and CSF, allowing us to optimize the TR/TE combination in order to create the most contrast between these three tissue types.

Neonatal development, however, makes it difficult to use adult optimized imaging parameters in infants primarily due to two distinct properties. First, the neonatal brain has a much higher water content at birth, with rapid decrease in water content in both grey and white matter as the brain develops. This causes the overall tissue to have longer relaxation times, and provides a moving target for optimizing sequence parameters. But more importantly, neonatal white matter is largely unmyellinated at birth. Rapid myelination occurs through the first year of life, creating localized regions of higher T1 signal starting with the corticospinal tract, followed by visual and inter-hemispheric connections, and ending with cortical association connections. As a result, neonatal development as observed by MRI can be roughly broken down into three distinct temporal stages [41]. The first stage is referred to as the infantile pattern (< 6 mo), and is approximately a reversal of the grey-white tissue contrast observed in the adult pattern (T1 and T2 are effectively switched). The isointense

period (between 8-12 months of age) is characterized by very poor differentiation between white and grey matter, with regions of higher T1 signal where myellination is occurring. And finally, the early-adult pattern ($>$ 12 months) is reached once the rate of myellination drops off, with tissue contrast more closely resembling a fully developed brain. All of this leads to sequence development challenges, requiring age specific optimization, as well as acquisition challenges, as smaller brains provide a lower signal to noise ratio (SNR) and neonates are more susceptible to motion in the loud MRI environment.

2.2.2 Substructure Segmentation

Brain segmentation, sometimes referred to as brain parcellation, is the process of discretely labeling a brain image into anatomically or functionally congruent regions. Classical parcellation methods involved the manual delineation of each region of interest, one subject at a time, by a trained user usually following a brain atlas guided protocol. Naturally, this was prohibitively time consuming for analyzing large cohorts, with each brain taking in the order of days to weeks to fully segment, depending on the granularity and number of regions desired. More recently, many attempts at automatic brain segmentation have been developed [42, 43, 44].

By and large, Freesurfer (FS) is the most widely adopted automated cortical [45] and subcortical [46] segmentation software in the neuroimaging community. FS uses two independent, but similar in approach, probabilistic pipelines for parcellating the cortex and subcortical substructures. One of the major challenges in automating brain segmentation is the inherent discord between geometrical structure and their conceptually generated labels. Structurally contiguous gyri and sulci may change labels as they cross cerebral lobes or perform different functions. Additionally, deep grey matter substructures may not have well enough defined tissue contrast at their border, or may be susceptible to partial volume artifacts due to their proximity to white matter or CSF. To attenuate these intrinsic properties in the segmentation algorithm, FS and most other methods currently published use a probabilistic atlas as a prior in conjunction with the local image intensity to classify each voxel into discrete regions. The FS atlas was created by transforming manually pre-

segmented subject brains into a common space and calculating the probability of each voxel belonging to each target class. This allows us to then transform our subject image into this probabilistic atlas space, and compute for each voxel the probability of it belonging to a particular class given both its new position in this standardized space and the original voxel intensity. The final step in the FS segmentation pipeline uses a Markov random field to model local spatial relationships based on intensity and class, improving the segmentation at tissue boundaries.

A problem arises when applying these existing implementations, which have been designed and optimized for fully developed brains, to a neonatal cohort. As described in the previous section, the developing brain has drastically different tissue contrast early on, and it undergoes rather significant changes over a very short period of time. This renders both the spatial and intensity priors used in the posterior calculation inadequate. The change in tissue contrast is significant enough that ideally we would require a different set of priors for each gestational week until development reaches the early-adult pattern. This is both labor and cost prohibitive. However, Gousias et. al have made publicly available a dataset, named ALBERT, of 20 manually segmented neonates ranging from 26 weeks gestation to full term [47]. The ALBERT subjects have been segmented into 50 cortical and subcortical substructures, and will serve as the set of templates to propagate the structure labels onto our subjects of interest. The benefit of having such a dataset available cannot be overstated, as generating such dataset in house would require an incredible amount of work. The authors state that for the 20 available subjects, a total of 18 person-months was required to generate all labels.

A promising implementation of automatic neonatal segmentation utilizing the existing ALBERT template has been proposed [48]. This work provides two possible methods for propagating the atlas labels into subject space: computing pairwise registrations between each ALBERT template and the target subject, or calculating one probabilistic atlas computed from all ALBERT subjects at once. In chapter 3 I will describe in depth how we have adapted this method to be more robust to both structurally abnormal subjects and less than ideal image quality, with the trade off of requiring a final manual correction step.

2.3 NEURAL NETWORKS

The concepts presented in this section will follow the notations and conventions presented in "Neural Networks and Deep Learning" by Michael A. Nielsen [49]. Neural networks have been around in one form or another in the machine learning world for decades, but only recently have gained widespread popularity. The main driving force behind this movement has been the increasingly low cost of entry as a result of cheaper and exponentially more efficient computational power. GPU computing has made it possible to run analyses that previously required access to very cost and labor prohibitive super computers. Increasingly complex problems and classic benchmarks have been improved upon by using variations of neural networks, from image classification on vast datasets to natural language processing and genomics [50, 51, 52, 53, 54, 55]. For a comprehensive survey of deep learning in medical imaging to date, see Litjens et. al.[56]

The basic idea behind neural networks, incidentally, is rather straightforward. A neural network is a hierarchical graph composed of individual units, called neurons, designed to apply a non-linear function to a weighted sum of the inputs, outputting a single value. Neural networks can be used with a variety of inputs, from gene expression data to speech patterns or free text. However, since the problem we are trying to solve in this work boils down to an image classification task, we will use two-dimensional images in the subsequent examples as a proxy for the segmented substructure images. Figure 4 shows the general structure of a neural network with a set of images as the inputs. The intensity of each pixel is multiplied by a unique weight, and the weighted sum of all pixels (assuming a fully connected network) becomes the input to the activation function. We will discuss how to find the optimal weights later in this section.

The choice of activation function is very important to both the final success or failure of the network, as well as the speed at which it learns. The earliest conception of a neural network attempted to use the step function for its activation. Equation 2.1 shows the general form of the step function, with its graph shown in figure 5. This is the classic perceptron, which outputs a binary 0 or 1, depending on whether the weighted sum of the inputs fall above or below a set threshold, which we call the *bias*.

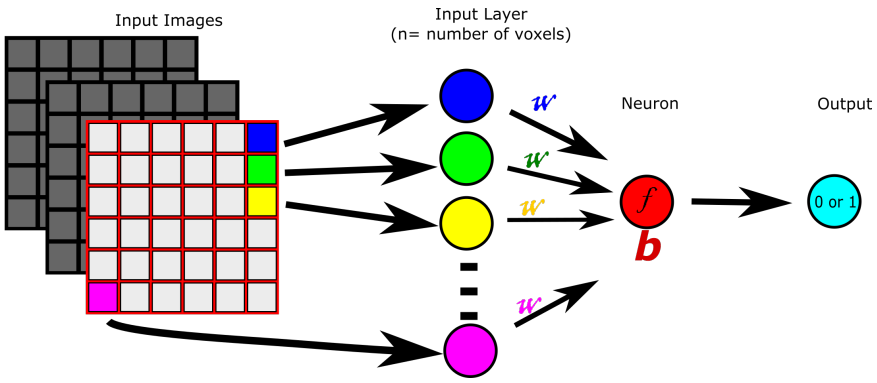

Figure 4: Basic Unit of a Neural Network

$$\text{output} = \begin{cases} 0 & \text{if } w \cdot x + b \leq 0 \\ 1 & \text{if } w \cdot x + b > 0 \end{cases} \tag{2.1}$$

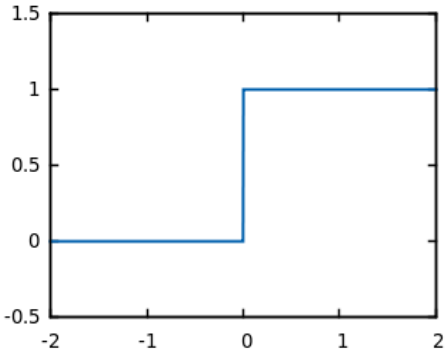

Figure 5: Step Function

There is a major shortcoming when using a perceptron as our basic unit in our network. As we try to learn the optimal parameters of our network, any small change in either the input or the weights we are learning can result in a polarizing change in the output. This can lead to an indefinite oscillation when learning the weights, especially when the calculated output hovers closely between the threshold, preventing the algorithm from converging.

One solution is to instead use a sigmoid function (equation 2.2). The sigmoid function, whose graph is shown in figure 6, has the benefit of allowing a smooth transition from 0 to 1, instead of hard binary outputs.

$$\frac{1}{1 + \exp(-\sum_j w_j x_j - b)}. \tag{2.2}$$

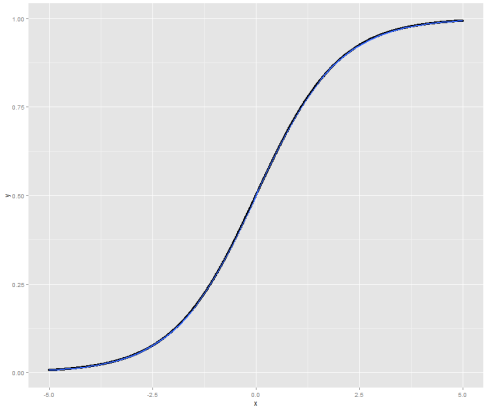

Figure 6: Sigmoid function

This allows us to make small changes in the weights, and observe a proportional change in the output. This property is what allows us to iteratively improve our predictions by making changes to the weights proportional to and in the direction of the gradient (see chapter 5 for a detailed description of the gradient descent algorithm).

Figure 7 now shows the basic neuron with a sigmoid activation function. This will be the basic building block for our network. Of note, a single layer network using a sigmoid activation function is equivalent to a multi-variate logistic regression.

While a single neuron is powerful, it cannot estimate non-linearly separable functions. Multiple neurons, however, have been shown to estimate any complex non-linear function (given enough neurons) [49]. This is where the power of the neural network comes from. By building a Neural Network comprised of multiple neurons, and possibly multiple layers of neurons, we can create layers of abstraction from complex data. Using more than one neuron in a single layer enables us to detect multiple features within the same level of

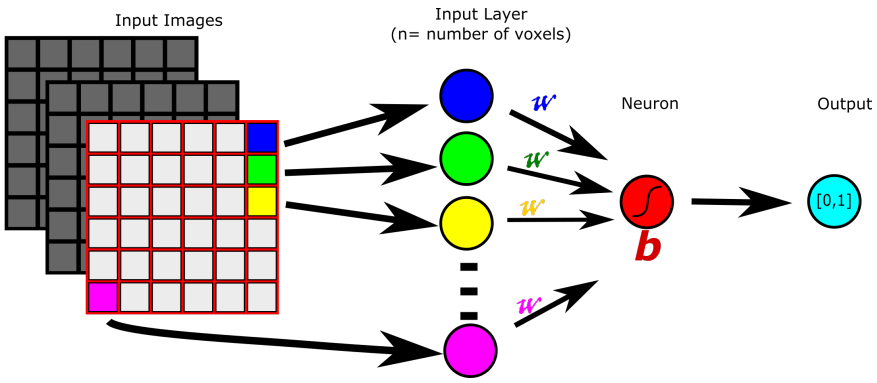

Figure 7: Single neuron with sigmoid activation function

abstraction (figure 8). Each neuron is tasked with detecting a specific signal that maximizes the activation of that particular function.

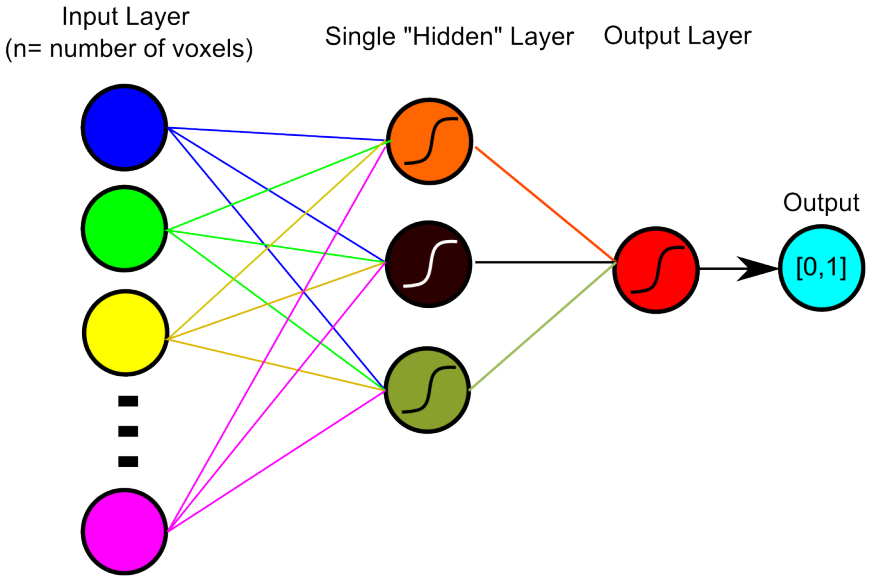

Figure 8: Single layer neural network

When we stack multiple layers of neurons, we call the middle layers "hidden layers" simply because we don't have direct access to their inputs or outputs. It is possible, however, to project their activations to give insight into how the features relate to the original input signal. We speak of data "abstraction" - because as we move deeper down our network, each

new layer's input is the output of the previous layer. This means that the hidden layers no longer look for features from the original raw input, but rather from the constructed activation signal from the previous layer. A simple (and somewhat idealistic) example comes from facial recognition. A deep neural network applied to facial recognition will use the first layer to detect simple abstract shapes such as straight lines or specific curves. Deeper layers will activate when these combined features form new, more complex shapes such as parts of a nose, eyes, or face. The final layers can then combine these high level abstractions to identify the presence of a face, or go even further and identify specific faces from the training set. There is no hard-set rule for where "Deep" Learning begins. The term has become much more relevant to marketing and promotion than to the actual architecture of a neural network, and generally 2 or more hidden layers can technically be classified as a deep network.

2.3.1 Convolutional Neural Networks

A fully connected network as described in the previous section does not take advantage of the intrinsic spatial information contained in images. Such network only learns patterns of activation as they appear in a fixed order of the training data. For example, a specific feature, say a triangle, that activates a neuron when present in top left quadrant of one training image would have to be encoded by a different neuron within the same layer if the same exact feature were present in a second training image, but instead located one pixel to the right. Furthermore, we "waste" neurons encoding sets of pixels randomly scattered across the entirety of the images, which are heuristically more likely to be noise and less likely to be useful in our classification task.

A proposed solution to these challenges are Convolutional Neural Networks (CNNs). CNN's use the concept of a Local Receptive Field (LRF) to "scan" the image and group only adjacent pixels into a neuron, rather than every pixel in the original image. This is able to identify position invariant, reusable features by having one full set of neurons in a hidden layer (called a feature map) share the same weights and bias. Figure 9 depicts the "scanning" method of a LRF for creating a feature map. This example uses a 2 x 2 LRF, where only

the pixels within the LRF contribute to the weighted sum of their designated neuron. The following neuron in the feature map then encodes the next 2 x 2 LRF, translated over by one pixel - and so on.

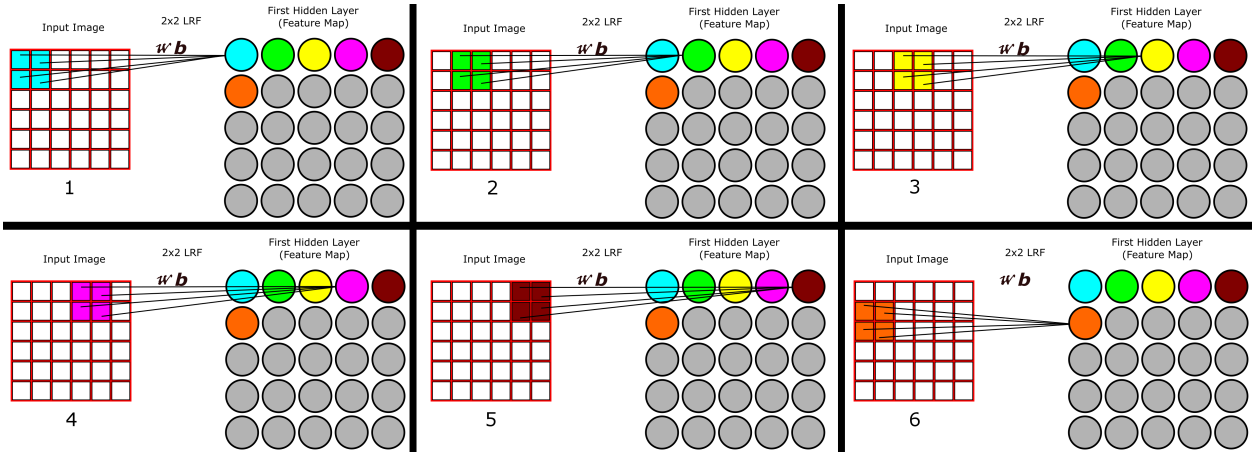

Figure 9: Local Receptive Field: First six neurons in feature map.

It is important to emphasize that the weights and bias are the same for all neurons in one feature map. This means that each neuron in the feature map activates when the same 2 x 2 (in this example) feature is present anywhere across the image. In short, a feature map, as its name implies, encodes not only the presence of a feature, but also its spatial location and number of occurrences within the input image. This has the added advantage of greatly reducing the number of parameters we have to estimate, as each neuron is only connected to a subset of input pixels, and we only have to learn one set of weights and bias for each feature map we construct. Similar to a fully connected network, we can then train multiple feature maps as part of each hidden layer, with each feature map encoding the spatial location of a specific feature, whether it's a simple shape or some higher level abstraction, present in the input image (figure 10).

Just as in a traditional NN, the feature maps in each layer can then be connected to subsequent layers of either additional feature maps or a fully connected layer. Optionally, the addition of a max pooling step following each hidden layer of feature maps has been shown to improve signal and further decrease the number of parameters to be estimated in each subsequent layer [57]. Max pooling works similarly to downsampling in image analysis,

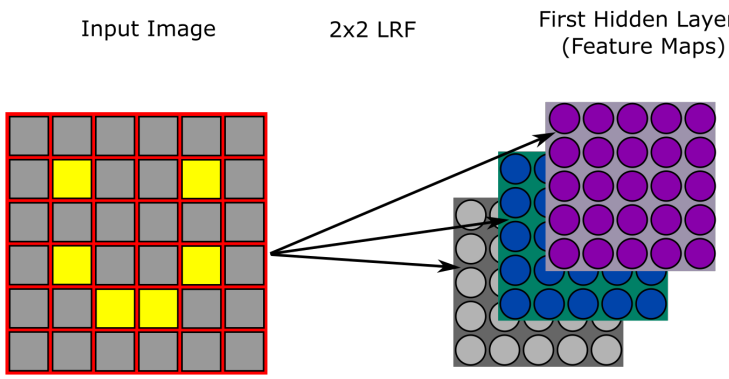

Figure 10: Multiple feature maps can be generated per hidden layer

where we reduce the image dimensions by taking the max value of a subset of activations, such as a 2 x 2 block of neurons. This has two advantages: one, it reduces the search space at each subsequent layer, and two, it smooths the data, effectively de-noising it and increasing our signal. A simple CNN would then look like figure 11. Note that the final output layer is fully connected to every neuron in each feature map in the previous layer.

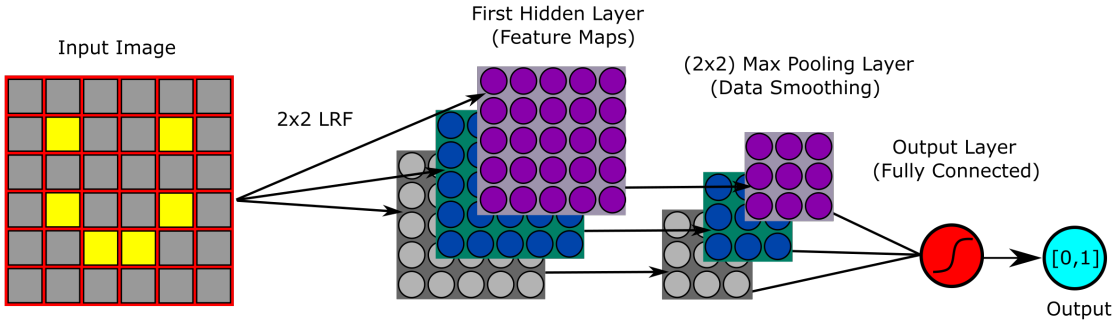

Figure 11: Single hidden layer convolutional neural network. Note that the final set of feature maps are fully connected to the output layer (not shown here for simplicity)

An example of a successful CNN is the LeNet5, published in a landmark paper [1] that is credited with the ressurgence of neural networks in mainstream AI. LeNet5 was shown to be effective in applications ranging from image recognition to text and document classification. For an in-depth survey of deep learning applications in medical image analysis, please see

the work by Litjens, et. al [56].

2.3.2 Learning the Weights

Learning the weights and biases of a neural network is done by an optimization algorithm known as Stochastic Gradient Descent (SGD). In this section I will provide a cursory overview of SGD and its application to neural networks. Please see chapter 5 for a more detailed description of how we compute and update the parameters at each learning iteration.

The first requirement in gradient descent is the selection of a cost function. This cost function is simply a measure of the accuracy of our classification task. Common cost functions used in neural networks include mean squared error (MSE), negative log-likehood, and cross-entropy. The negative log-likelihood cost function is particularly powerful in conjunction with a final softmax activation layer, and is typically the go-to function in CNNs. The softmax activation function, shown in equation 2.3, normalizes the output of each neuron of the final layer in our network over all possible output neurons, which are the desired classifiers.

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad \text{for j} = 1, ...,K \tag{2.3}$$

Where z is is the weighted sum of the outputs of the previous layer, and K is the total number of neurons in our final layer (the classifiers). The output of the softmax for each neuron (j) in the final layer is then the likelihood of the given input being classified as that particular label.

Intuitively, we tend to think in terms of minimizing our cost function, therefore we use the negative log-likehood function, which as its name implies, takes the natural log of the softmax output for the correct classifier (to facilitate with differentiating when calculating the gradient), and negates it (in order to minimize it). Minimizing this cost function is equivalent to increasing the prediction accuracy of our network.

Gradient descent works by calculating the gradient of the cost function given the current parameters of the network, and then changing these parameters in the opposite direction

of the gradient, scaled by a small factor - called the learning rate. Appendix A goes in further detail on how we calculate the gradient of our network using the backpropagation algorithm. The gradient of the cost function is analogous to its slope, generalized to more than two dimensions, and indicates the direction in which the function is increasing. This makes it easy to see that if we move in the opposite direction of a function's gradient, we move towards a minimum point in the function. We have no guarantee that the minimum we reach is global or local, especially in very high dimensional functions. However, we can mitigate this by running multiple instances of learning with randomly initialized weights. Additionally, techniques such as layer drop out and regularization help us further attenuate this problem.

Calculating the gradient using hundreds or thousands of input images can quickly become computationally expensive. We can overcome this by instead calculating the gradient using only a subset of our input at a time. These subsets are called mini batches and are randomly selected from the full input dataset. This is where the term stochastic gradient descent is derived from. By enforcing our network's cost function to be an average over all individual input cost functions, we allow ourselves the benefit of being able to iterate over our entire dataset in batches, calculating the gradient from this subset of examples at each turn. Once we have iterated over all inputs once, called an epoch, the resulting cost (and resulting change in weights based on its gradient) will be equivalent to having done so using the entire dataset at once. This feature of neural networks enables us to train over extremely large datasets while avoiding hardware limitations when computing the gradient of our chosen cost function.

As mentioned earlier in this chapter, we chose the sigmoid function as our activation function because it provides more granular feedback on the output of each neuron when we make small changes to the weights as we train our network. As we alter the weights according to the gradient, we observe a proportional change in the cost function, allowing us to iteratively improve our classification accuracy as we learn from more training examples. One caveat exists for this function, however. Observing the shape of the function, we can intuitively imagine the rate at which we learn our weights is dependent on the slope of the function. This works to our advantage when the input to the function lies within the middle

portion of the graph, where we will observe a rapid change in our output as our weights change accordingly. However, if the input to the activation function lies at either extreme of this graph, the rate at which we learn our weights, which is dependent on the slope, will suffer from significant slow down. In practice, we can leverage alternative activation functions that will behave in a better fashion when the input falls at either extreme. The hyperbolic tangent function (tanh)

$$tanh(z) = \frac{e^{2z} - 1}{e^{2z} + 1} \tag{2.4}$$

has a similar shape to the sigmoid function, however it is smoother at the extremes, reducing learning slow down. One particularly powerful and simple activation function that has become very popular in neural networks is the rectified linear units (ReLU - equation 2.5).

$$f(x) = max(0, x) \tag{2.5}$$

The graph of this function is shown in figure 12. As we can see, this function prevents learning slow down by monotonically increasing directly in response to the input while attenuating the effects of low extremes by reducing them to zero. This gives us a more direct measure of the strength of activation at each neuron, and still satisfies our cost function requirements when used in conjunction with a final softmax layer. ReLU, however, is not without its pitfalls. It is highly sensitive to the initialization parameters, and in the presence of sparse datasets, may lead to a problem known as the "vanishing" gradient, where the updated weights drop to zero without chance of recovering, and "exploding" gradients, where a run-away effect causes the weights to continue to exponentially increase. Exploding gradients can lead to overflow errors and non-generalizable parameters.

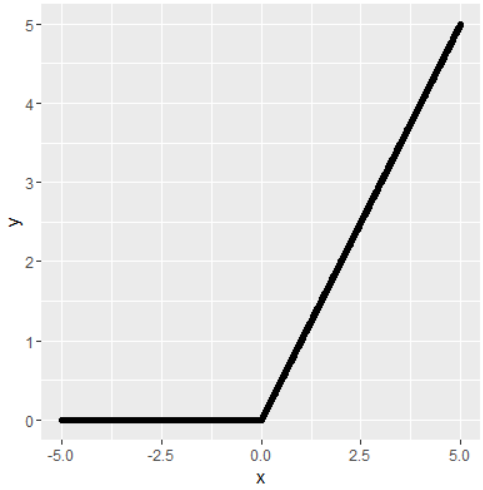

Figure 12: Rectified Linear Units Function

2.3.3 Model Creation

There is no closed form method of designing an optimal neural network. Choosing the hyper parameters, such as learning rate, number of hidden layers, number of neurons or feature maps per layer, activation function, etc... can become as much of an art as it is a science, requiring trial and error and patience. There are, however, heuristic approaches that we can use in order to expedite our search for an effective architecture. Before committing to fully training a specific network architecture, which can take weeks or months depending on the size and complexity of the dataset, we can instead perform quicker cycles of learning on a subset of the data without reaching convergence or saturation. We can apply a combination of random and grid search to cycle over viable parameters [57], giving us an estimation of what hyper parameters are likely to work, and getting closer to an "optimal" network. The nature of deep learning makes it so that any given network is never optimal, as there exist an infinitely large number of networks that may perform equally or better. Instead, we strive to design an effective network, and iteratively try to improve on it.

A grid search approach to identifying hyper parameters requires the user to explicitly choose sets of hyperparameters to iterate over in order to find the best set. Conversely,

random search sets a range of possible values for each hyperparameter, and randomly samples from this range at each iteration. Combining random and grid search gives us the most power, where we can specify a larger restricted set of possible ranges of hyperparameters at a more automated level. When choosing hyperparameters, it is important to avoid an overly complex model in order to improve its ability to generalize to new, independent datasets. The following section will describe strategies for overcoming this challenge.

2.3.4 Training and Validation

The task of any classification algorithm in machine learning is to learn parameters that correctly classify the inputs into the desired classes. However, the larger goal of these methods is to be able to generalize these parameter to broader, independent datasets. If the algorithm is only good at classifying our specific dataset, for which we already know the labels, then it is effectively useless.

It would be problematic if we only measured our accuracy on the same dataset we used to learn the parameters, as we would have no information on how these parameters would generalize to new inputs. In fact, if we only measure our accuracy on the training set, it leads to a problem known as overfitting. Overfitting is analogous to our learning algorithm "memorizing" the training data, rather than "learning" the parameters that are effective at classifying them. If we let our learning algorithm run indefinitely, it is likely that we would reach 100% predictive accuracy in the training data, especially the more complex our model is. But this would not reflect how the algorithm would perform on an independent dataset. In order to verify whether our chosen hyperparameters are in fact learning to correctly classify its inputs, we need to measure its predictive accuracy on an independent dataset at each iteration. To counteract overfitting, we can split our dataset into two independent sets: the training set and the validation set. The training set is generally the largest portion of the data, with guidelines suggesting around 70-80% of the total data, and provides the network with inputs with which to learn the parameters - in our case the weights and biases. At the end of each training epoch, we use the validation set to measure the improvement the algorithm has made in the classification task. Figure 13 shows a classic example of the

behavior of overfitting on a learning algorithm that is left to run past its optimal stopping point.

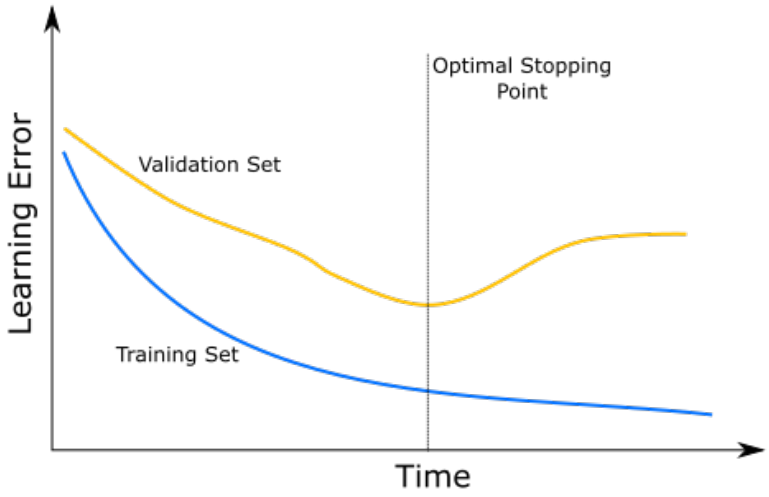

Figure 13: Example of overfitting in machine learning

While we continue to improve the algorithm's performance on predicting the training set labels, it loses its ability to generalize to an independent dataset (the validation set). This happens because over time more and more of the improvement of the algorithm can be attributed to the idiosyncratic attributes of the individual examples in the training set, and less to the general properties of the overall dataset.

Neural networks are extremely powerful, but as most machine learning methods, they work best with large datasets. Therefore, we are reluctant to give up such a large portion of our precious data to simply validating our model. We can compromise on this by using a method known as K-fold cross validation. In K-fold cross validation, we split our data into K sized independent sets. We use K-1 sets as the training set, and the remaining dataset becomes the validation set. We then repeat the learning process K times, each time using one of the remaining independent sets as our validation set. This ensures that every example is used in learning at least once, and each time the parameters are validated against an independent set. Once we are satisfied with a model with a satisfactory predictive accuracy

both training and validation sets averaged over each of the K runs, we can then use the entire dataset to train our final algorithm. If our dataset is large enough, it is advantageous to further split the dataset into an additional test set. The test set is usually about the size of the validation set, except it is never used to learn parameters. It is used as a fully independent dataset to test the final predictive accuracy of the learning algorithm. The gold standard for test sets is generally to use an entirely new dataset unrelated to the training and validation sets.

2.3.5 Existing software implementations

Neural networks have become increasingly accessible due to the availability of cheap and efficient hardware. This has spurred the development of several frameworks created with the explicit purpose of generalization and optimization of neural networks. One of the original frameworks is called Theano [58], and is likely the most widely adopted frameworks to date. Theano was created as a Python interface for symbolic math, built on top of the excellent numpy library. Its power, however, comes from C language optimization and GPU computing integration. Theano aims to provide a simple interface for designing neural networks, but with highly optimized code doing the heavy lifting. With the success of Theano, Google has decided to develop their own implementation of a very similar idea with TensorFlow [59]. TensorFlow is still in its infancy, with many features still missing and not quite reaching Theano level benchmarks, but with the power of Google behind it, is already on its way to becoming the major player in deep learning development. Another framework worth mentioning is Torch [60], used by large enterprises including Facebook and Twitter. Torch is more specialized than Tensorflow and Theano, in that it is highly optimized for running on NVIDIA GPU's. It is a C/CUDA library with a scripting interface built with LuaJIT. It is less flexible and extensible than the previous two, and is more directed at specialized commercial and enterprise level applications.

3.0 STRUCTURAL SEGMENTATION METHODS

3.1 OVERVIEW AND CLINICAL APPLICATION

Brain segmentation is the process of discretely labeling a brain image into anatomically or functionally congruent regions. While many robust methods exist for automatic segmentation of fully developed brains, neonatal segmentation methods have yet to achieve similar performance. Existing attempts at automated neonatal segmentation fall short in both segmentation accuracy and the quality of data that they are capable of processing. In clinical-translational applications, we do not always have the luxury of having high quality imaging of healthy controls. Brain injury that leads to morphological variance, motion artifacts, and cross-scanner variability when analyzing data acquired across institutions are just some of the problems encountered when analyzing clinical datasets. When facing these challenges, many research labs end up relying on manual delineation methods, which are tediously laborious and can suffer from lack of reproducibility. In this chapter, I describe the Neonatal Brain Structure Segmentation (NeBSS) pipeline, developed to be a robust alternative flexible enough to handle clinical data where existing automated or semi-automated implementations are too rigid to successfully process. I will first give an overview of the computational tools used in the pipeline, followed by a description of the full processing workflow.

3.2 COMPUTATIONAL TOOLS

3.2.1 BET

Extracting the brain from the remaining anatomy is not a trivial task. However, removing the extraneous tissues such as bone, eyes, and neck, is a necessary step in many neuroimaging algorithms, as they often introduce noise and heterogeneity to the input data. This problem is exacerbated in neonates, as the field of view with respect to the head is often larger than a typical adult brain imaging protocol, and the inclusion of shoulders and even hands in the image is not uncommon. Several tools exist for brain extraction, but FSL's Brain Extraction Tool (BET) [61] is the most widely used due its robustness and fast processing speed.

Developed by Stephen Smith over a decade ago, BET uses a combination of intensity histogram and surface modeling to generate the brain surface. The algorithm is an iterative procedure that first uses the intensity histogram of the image to find the brain's "center of gravity." This position in the image is then used to initialize a tesselated sphere roughly half the size of the estimated whole head volume. The algorithm then iteratively allows this sphere to grow one vertex at a time, expanding outwardly from the center of the brain. The following three modeling constraints are placed in the expansion of the tesselated sphere to optimize the algorithm's performance. A within-surface vertex spacing constraint is placed to ensure that the sphere grows uniformly outwards from the center of the gravity. This contraint can be altered manually if necessary in cases where the center of gravity is not well estimated or an intensity gradient artifact is present in the image. Secondly, a surface smoothness control parameter is imposed to reduce jagged edges and ensure a smoother final brain extraction. Finally, a third parameter is used as a fractional threshold d that is used as an approximation of the intensity difference between voxels deemed inside the brain and outside. In other words, this is used to determine the difference in voxel intensity between the surface of the tesselated sphere and the outermost voxel at which the sphere stops growing. Lower values of this threshold give more conservative estimates of the brain surface and may result in a brain extraction that includes voxels outside of the brain. Figure 14 shows the result BET applied to a neonatal brain MRI. We see that although the algorithm

performs very well, some skull is included in the final brain extraction, at the level of the superior parietal cortex. From our experience, a small amount of skull remaining in the brain extraction does not interfere with the downstream elements of the pipeline.

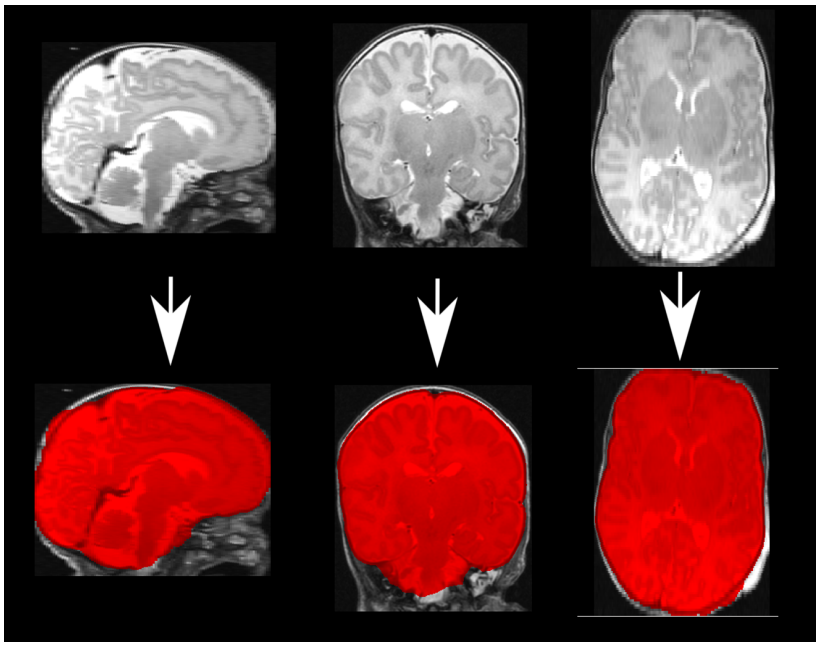

Figure 14: FSL Brain Extraction Tool (BET) Applied to a Neonatal Brain

3.2.2 Bias Correction

A bias field is a common artifact encountered in MR Imaging that exists as a result of inhomogeneities in the RF field of the magnet. A bias field artifact is generally consistent within a particular scanner, but it is sensitive to the subject's proximity to the head coil and other external components. The bias artifact presents as a low spatial frequency intensity gradient, usually along a particular axis of the imaging volume. Figure 15 shows an example of a bias field artifact in a neonatal brain, followed by a bias field correction. The artifact is particularly prominent along the Anterior - Posterior axis, with the frontal regions significantly darker than the posterior regions. This intensity gradient becomes a nuisance for image registration algorithms that use local and global intensity values to co-register two images, as well as tissue segmentation applications that use global voxel intensities to

cluster them into estimated tissue classifications. In our pipeline, we use FSL's built in bias correction tool [62, 63] to normalize the brain image following the brain extraction step.

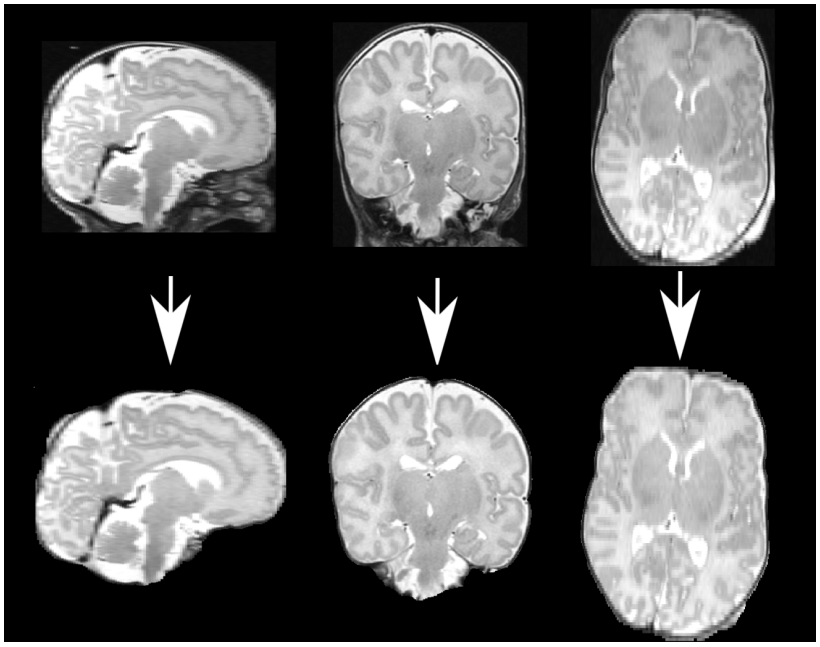

Figure 15: FSL Bias Correction Tool Applied to a Neonatal Brain

FSL's method uses a modified Expectation-Maximization algorithm developed by Wells et al.[64] that couples the estimation of the bias field to intensity-based tissue segmentation. The intuition behind this method is that in order to accurately perform an intensity-based segmentation, it is necessary to correctly estimate the bias field. However, if you know the bias field, then tissue-based segmentation becomes a trivial task. By coupling these two problems and modeling them as Gaussian distributions, we can use a maximum *a posteriori* probability (MAP) criterion to estimate them. The EM algorithm iterates between modeling the bias field as a gradient vector across the 3D volume, and modeling a 3 compartment intensity based tissue classification. At convergence, the bias field is estimated and the tissue classes are approximate to empirically derived estimates. While this method performs well for tissue classification in developed brains, due the varying degrees of myellination through the first year of development, it falls short in neonatal analysis. The bias field, however, is adequately estimated.

3.2.3 ANTS

ANTS is a symmetrical, diffeomorphic registration algorithm that continues to gain popularity in a plethora of neuroimaging (and other) applications [65]. A diffeomorphic registration maps an input to a target, where each point in the input has a one-to-one equivalent in the target. This is in contrast to elastic transformation algorithms, in which this property does not hold, making them inconvenient when mapping pre-defined segmentation labels between two spaces. A symmetric transformation means it has an invertible function, allowing us to both map our target to a template, as well as mapping the template back into the target. Figure 16 shows intuitively the power of a symmetrical diffeomorphic algorithm, when in conjunction with a linear (affine) registration can seamlessly map two morphologically different images symmetrically to one another.

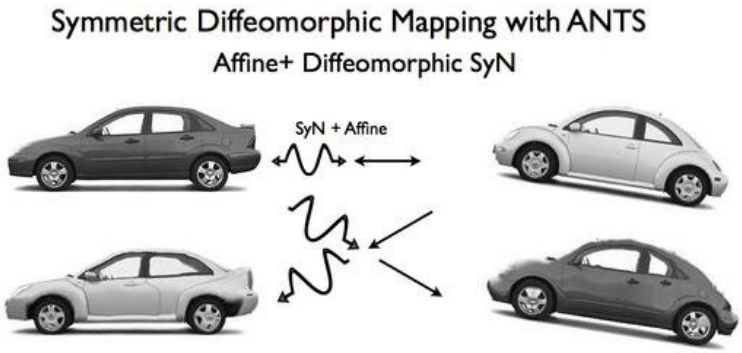

Figure 16: SyN Function used in symmetric diffeomorphic mapping with ANTS [from Avants 2008 [66]]

The power of ANTS comes from its novel image normalization method, termed Symmetric image Normalization (SyN), which maximizes the cross-correlation within the space of diffeomorphic maps. Traditional elastic registration methods, in contrast, seek to balance a regularization and similarity term within deformable parameters based on domain specific constraints. These methods tend to fail, or not go far enough, depending on the dissimilarity of the input images, and are generally highly sensitive to the regularization parameters. ANTS bypasses these constraints by performing the optimization within the transformation

space, rather than the original image space. Given enough time, ANTS is guaranteed to find a local maximum in the similarity metric (however, it is still subjective to local maxima). While a significant improvement in image registration, ANTS still requires careful consideration of the initialization parameters and similarity function chosen. Highly dissimilar images, or noisy data, can greatly increase computational time and increase the time to conversion exponentially. As such, in this work we do not deviate from the recommended optimal parameters that have been empirically tested by the ANTS development team.

3.2.4 ALBERT Templates

Automated structure segmentation algorithms in general rely on calculated tissue priors[45, 67] generated from a pre-segmented subjects, or existing anatomical atlases[68]. One way or another, a pre-existing manually segmented dataset is a necessary first step. This becomes a prohibitive barrier to entry, as generating an in-house atlas is a very laborious task. Fortunately, Gousias et al. have published a publicly available dataset, named ALBERT, of T1 and T2 images generated from 20 neonatal subjects [47]. The ALBERT subjects have been segmented into 50 cortical and subcortical substructures, and will serve as the set of templates to propagate the structure labels onto our subjects of interest. The authors state that for the 20 available subjects, a total of 18 person-months was required to generate all labels. Figure 17 shows a sample of 10 segmented ALBERT subjects.

The dataset consists of 15 preterm born neonates (8 female) scanned at term equivalent age with a median gestation age of 29 weeks (range 26-36 weeks) and median post-menstrual age at time of scan of 40 weeks. Additionally, there are 5 term born neonates (two female), with a median post-menstrual age at time of scan of 41 weeks (range 39-45 weeks). This enables us to select a subset of templates that more closely match our target subject to improve the registration accuracy.

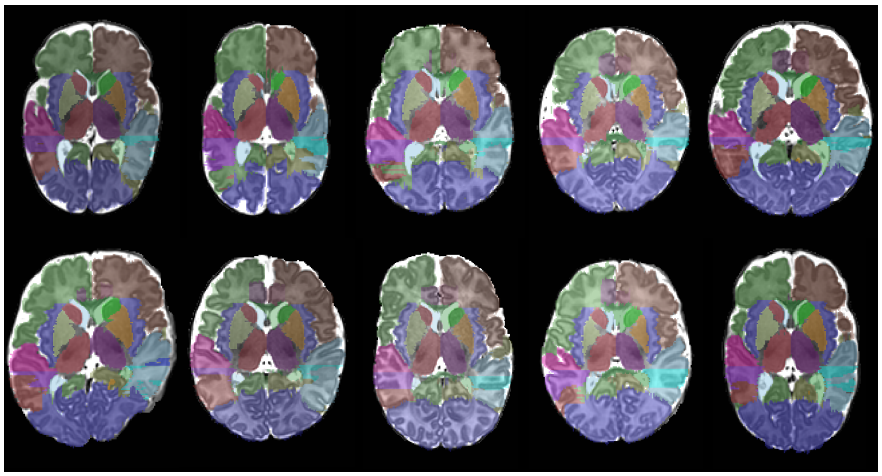

Figure 17: Sample images of 10 ALBERT template subjects

3.3 PROCESSING PIPELINE

3.3.1 Pipeline Overview

We have developed and implemented a semi-automated neonatal brain segmentation pipeline, named NeBSS, summarized in figure 18. NeBSS implements a variation of template based brain segmentation that uses a non-linear transformation algorithm to register a set of pre-segmented template brains into the subject's space.

The first step in the pipeline is to pre-process the input MRI in order to remove unnecessary tissue and noise prior to the registration. We first run a brain extraction with BET. This isolates the brain at the skull, but preserves extra-axial fluid, which is informative in different clinical contexts. Next, we use FSL's built in bias correction tool to normalize the brain image, removing any intensity gradient artifact due to field inhomogeneity. We are left with an intensity normalized, clean subject brain still in the native scanner space, retaining its original volume and proportions. This is the input to the registration steps. As figure 18 shows, at this point the pipeline is effectively split into two independent, parallel branches. We have previously introduced the ALBERT neonatal parcellation dataset [47] (branch A), which contains 20 pre-segmented neonatal brains of varying gestational ages. This dataset

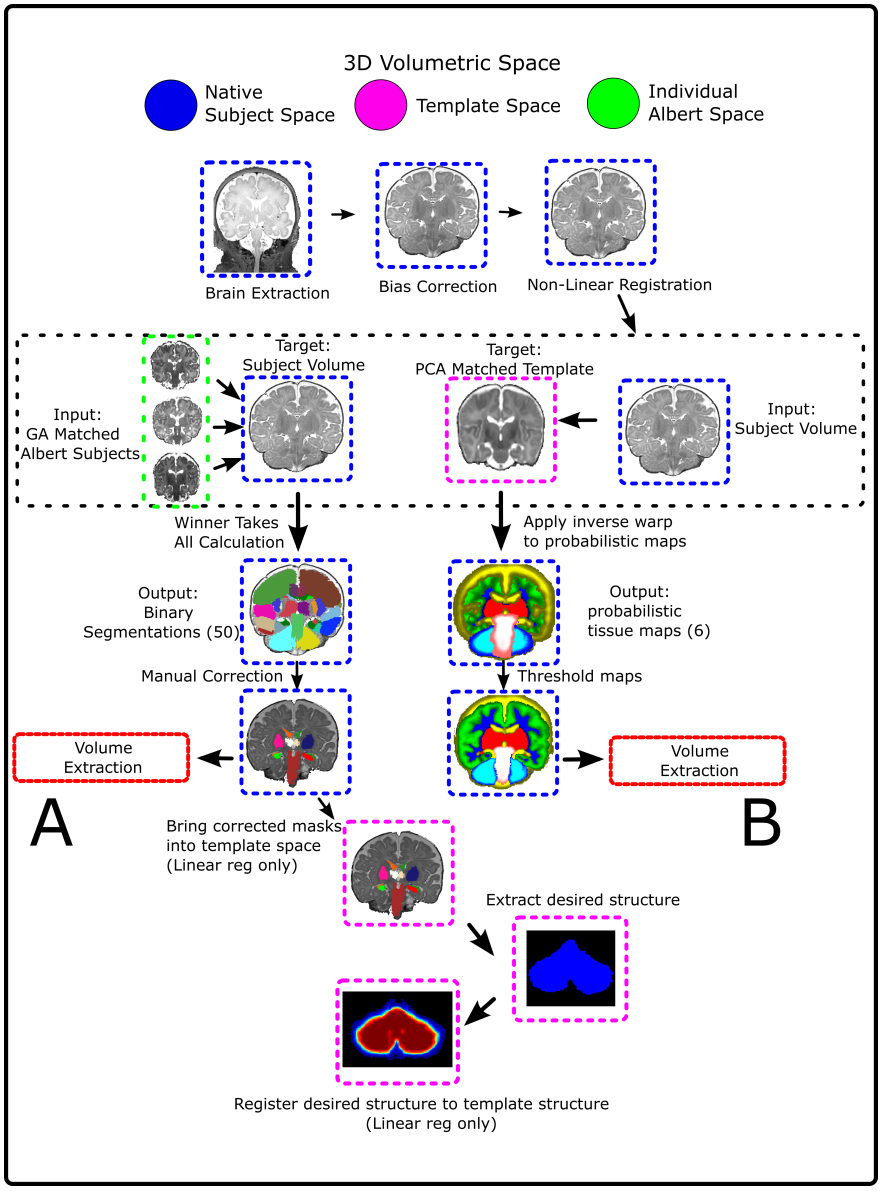

Figure 18: NeBSS: Neonatal Brain Structure Segmentation Pipeline.

delineates 50 brain substructures per subject, which we can use to propagate onto our subject through image transformations. However, at times specific clinical questions require as input additional substructures which are not present in the ALBERT templates, such as extra-axial fluid and total brain volume (for normalization). Furthermore, our previous work has also required global tissue volumes in neonates, such as total white, gray, and CSF volumes. Our solution to this is to use a probabilistic neonatal atlas created by Serag et al. [69]. This atlas contains gestational age specific probability maps of the following gross tissue volumes: CSF, cortical grey matter, deep grey matter, white matter, brainstem and cerebellum. We further modified the CSF probability maps to differentiate supra-tentorial, infra-tentorial and intraventricular CSF. This is incorporated into branch B of the pipeline. Both branches use the Advanced Normalization Tools (ANTS) algorithm [70, 71] to calculate a non-linear transformation between a target image and the source image.

Branch A uses ANTS to calculate a non-linear transformation of the pre-parcellated ALBERT subjects into native subject space. We choose four ALBERT subjects closest in gestation age to our subject to increase our registration accuracy and yield segmentations that more closely match the developmental stage of the subject. We then perform a voxelwise winner-takes-all calculation across the four transformed ALBERT subject labels to determine the substructure classification at each voxel of our subject. If there is a tie, we randomly select one of the competing labels. We are left with 50 non-overlapping discrete regions mapped onto our subject space. Finally, before we extract our features, a manual correction step is necessary to ensure anatomical accuracy of our label propagation. If we start with a good quality image, minimal manual correction is needed. However, severely abnormal brains or poor data quality will require more laborious manual correction. Once we are satisfied with the segmentation, we extract the substructures of interest and calculate their volumes. Since we are only concerned with classifying dysplasia in the second portion of this study, we need to decouple the volumetric information from each substructure. We achieve this by linearly transforming each substructure onto a standardized template space. This has the additional effect of reorienting all subject substructures onto a common space, potentially increasing our input signal by removing any large variation in both position and orientation.

Branch B works in the reverse direction as branch A. We non-linearly transform our subject brain into the gestation age matched probabilistic atlas space. This is done because the higher resolution atlas provides a better registration target, particularly when dealing with lower quality images. The resulting transformation is then inversed, a convenient feature of the ANTS algorithm, allowing us to bring the tissue probability maps into the subject's native space. This gives probability maps for each tissue type in the atlas in our subject's space. The final tissue volumes are calculated by binarizing each tissue map at a user defined probability threshold. Higher quality input images yields higher probability thresholds, giving us more confidence in our final volume. Finally, since each branch of the pipeline outputs a cerebellar volume, it presents a convenient tool for quality control and validation. We are able to calculate the precision of each branch of the pipeline by directly comparing cerebellar volumes, using the final manually corrected cerebellum from branch A as the gold standard.

3.3.2 Software Implementation

This pipeline is written in Python, and is structured using the Nipype framework [72]. Nipype is an open source, neuroimaging specific processing pipeline engine designed to create transparent and reproducible data processing graphs. Nipype contains built-in interfaces to the vast majority of existing neuroimaging software libraries, including FSL and ANTS. Additionally, Nipype provides tools for extending their library and creating project specific classes and interfaces. One of the biggest strengths of Nipype's pipeline engine is the exhaustive attention to efficient reproducible work. File checksums are calculated prior to each node in the processing graph, allowing us to rerun any step in the pipeline without having to compute all the previous nodes. This is achieved by verifying the checksum of the inputs at each node. If they match the previously computed files, that node will be skipped and only nodes containing a change in the input have to be run. All code created in this work is open source, published under the BSD License, and available at: https://www.github.com/PIRCImagingTools/NeBSS

3.3.3 Workflow

The structural MRI can be fed into the pipeline directly from acquisition. The only inputs required from the user are the subject's gestation age, post-menstrual age, and the coordinates for a bounding box in order to crop excessive extra-cranial structures from the input image. This last parameter is necessary when imaging neonates, as we don't always have control over what position they are in; as long as they are comfortable and sleeping. Because of this, often the field of view ends up capturing the infant's arm and shoulders. This can be problematic for the brain extraction and registration steps. Figure 19 shows the graphical user interface for the structural segmentation pipeline, named NeBSS.

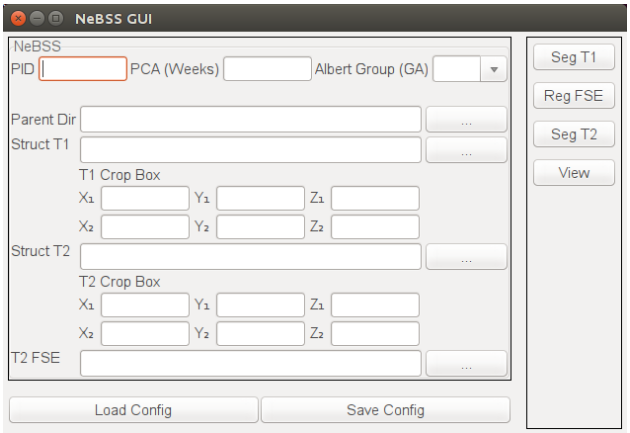

Figure 19: Graphical User Interface for the Neonatal Brain Structure Segmentation (NeBSS) Pipeline

NeBSS works optimally with T2 weighted images, as they have much better tissue contrast in neonates, especially grey-white matter differentiation. Volumetric T2 imaging, however, can be more susceptible to motion artifacts compared to T1 weighted images. As a compromise, NeBSS provides the user an alternative. We can instead acquire motion corrected T2 images, such as fast spin echo (FSE), up to 2 mm slice thickness. By registering this image to a volumetric T1, we retain the volumetric accuracy while leveraging the better tissue contrast from the T2 image.

Once NeBSS finishes running, the user is required to verify the label registrations, and perform any manual corrections if necessary. Any imaging specific software can be used

for this step, but we recommend ITK-SNAP [73] as it is purpose built for volumetric label creation and editing. Figure 20 shows a screenshot of manual correction of the output labels using ITK-SNAP.

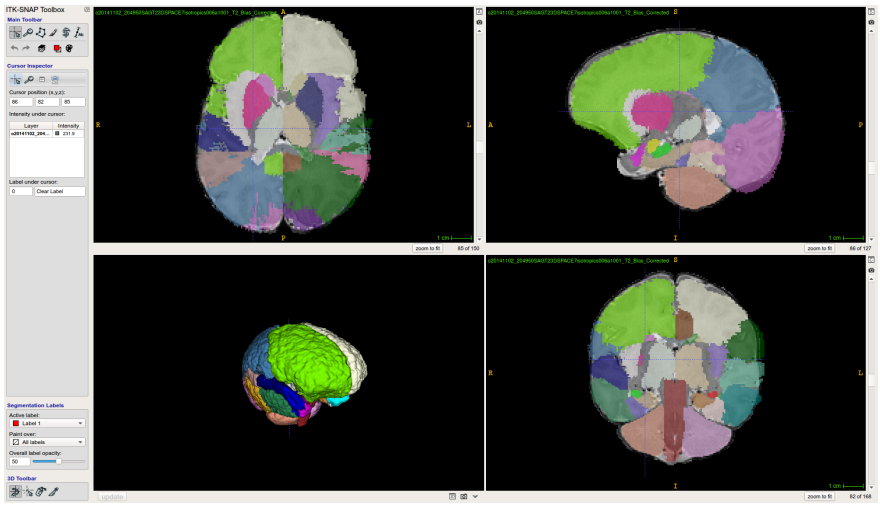

Figure 20: Example of manual correction interface using ITK SNAP

The segmentation pipeline outputs 50 different delineated brain substructures which can be used in additional applications, as well as global white matter, grey matter, and CSF volumetric measurements. See figure 21 for a sample output of the tissue probability maps. Given a neonatal image without severe motion artifacts and good tissue contrast, the total processing time can be as low as 5 hours, including one hour dedicated to manual segmentation of the desired substructures.

The processing time, however, can vary significantly based on the original image quality. While the pipeline is robust and is capable of processing rather low quality images, significant processing time is added. This is due to both the time it takes for the registration algorithm to converge, as well as additional time required for manual segmentation. Further evaluation and optimization of this workflow will be discussed in chapter 4.

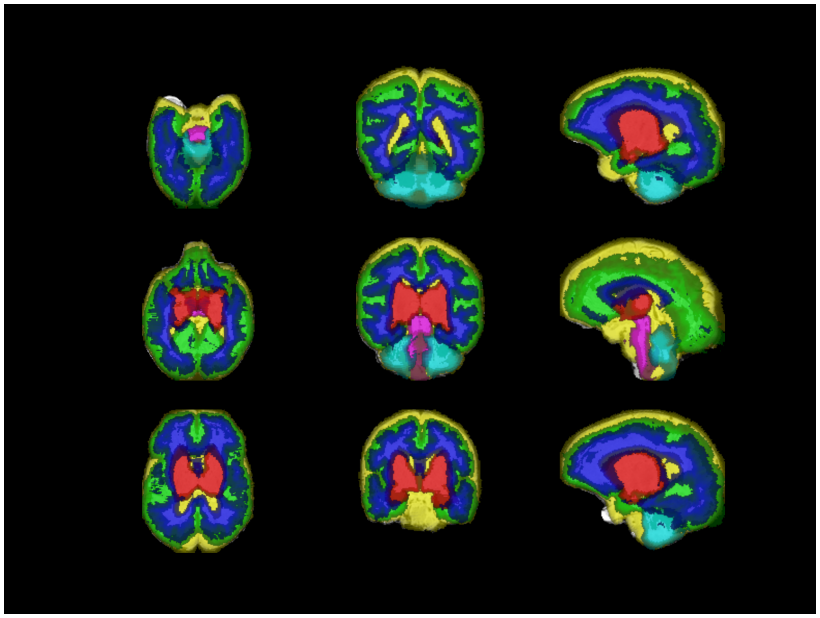

Figure 21: Example of Probabilistic Maps Output

3.3.4 Evaluation

We prospectively recruited 62 term CHD neonates, 54 term healthy neonates and 51 preterms neonates, all of whom underwent brain MRI with volumetric imaging. Mean gestation age (GA) for the preterm neonates was 30.7 (+/- 3.8) weeks and post-menstrual age (PMA) at time of MRI was 40.5 (+/- 7.8) weeks. Mean GA for the term neonates was 41.2 (+/- 3.8) weeks and post-menstrual age (PMA) at time of MRI was 43.5 (+/- 5.5) weeks. Mean GA for the CHD cohort was 38.0 (+/- 2.9) weeks and mean PMA at time of MRI was 42.4 (+/- 6.9).

The accuracy of a structural segmentation can be measured by comparing the output to a known gold standard. We can use the Dice Similarity Coefficient (DSC)[74] as a metric:

$$DSC = \frac{2|X \cap Y|}{|X| + |Y|} \tag{3.1}$$

Where $X \cap Y$ is the number of voxels the two images have in common, and the denominator is the total number of voxels in each structure. This is a simple measure of similarity that

will allow us to compare the raw NeBSS output to the manually corrected substructures - our gold standard. By calculating this measure for every subject in our datasets, we may be able to identify potential sources of segmentation errors, such as specific protocol or artifact, and allow us to further improve the segmentation algorithm. To avoid segmentation errors from propagating into the classification task, we will use the manually corrected segmentations as the training dataset for the convolutional neural networks.

Moreover, as the output of our pipeline is highly dependent on the accuracy of the substructure extraction, we must test our reproducibility, using the DSC to measure inter- and intra-rater reliability. We assessed the inter-rater reliability by having two independent users perform manual correction on the same set of six subjects (3 Controls and 3 CHD). For intra-rater reliability, one user performed a repeated round of corrections on the same set of subjects.

3.3.5 Clinical Translational Application

Finally, we applied this pipeline to assess volumetric differences in a neonatal cohort of term born neonates in comparison to healthy, term born controls. This work is done as part of a larger multi-disciplinary study in collaboration with developmental biology to assess cross-species phenotypes of brain abnormalities in human and mice with CHD.

We prospectively recruited 72 term born neonates diagnosed with CHD, and 32 healthy, term born neonatal controls, who underwent brain MRI with volumetric imaging. There is some overlap with the subjects used in the evaluation component in the previous section. Mean gestational age (GA) for the term neonates was 40.3 (+/- 2.9) weeks, and post-menstrual age (PMA) at time of MRI was 44.7 (+/- 5.7) weeks. Mean GA for the CHD cohort was 39.1 (+/- 1.9) weeks and mean PMA at time of MRI was 42.4 (+/- 4.1). Cerebellar and hippocampal volumes were extracted using Branch A of the segmentation pipeline described in the previous section. Structural volumes were PMA adjusted and compared across cohorts, with a threshold of < 52 weeks PMA. Statistical analysis was performed using ANOVA and Tukey's HSD test where appropriate, in R [75].

4.0 STRUCTURAL SEGMENTATION RESULTS

4.1 EVALUATION

4.1.1 Segmentation Accuracy

Comparing the Dice coefficient between the raw pipeline output and the manually corrected substructures gives us an estimate of the segmentation accuracy of the pipeline. We selected the cerebellum and hippocampus as representative substructures to perform the analysis. Figure 22 shows the Dice coefficients between the pipeline output and manual correction for subjects in each cohort. The cerebellum showed higher Dice coefficients, indicating relatively less manual correction was necessary compared to hippocampus. The Control subjects required less correction than both CHD and preterm cohorts ($p < 0.001$ for Cerebellum, $p < 0.025$ for Hippocampus), likely due to fewer structural abnormalities and more consistent imaging protocol. Additionally, we see a higher variance in the Dice coefficients in the two abnormal cohorts when compared to controls. This is summarized in Table 1.

To explore the reason behind the higher variance in Dice coefficients between raw output and manual correction, in particular for the abnormal cohorts, we visually inspected the subjects with the lowest values. Figure 23 shows sample images from the subjects with the lowest Dice coefficient, irrespective of cohort. The severely asymmetric and dilated ventricles have traditionally been a challenge for registration based segmentation methods. In particular, a substructure like the hippocampus, which closely snakes adjacent to the lateral ventricles is highly susceptible to misregistration errors. Furthermore, despite the exceptional performance ANTS has proved in its recent implementation, severe asymmetry is still a challenge. Additionally, noise, motion, and severe inhomogeneity artifacts can reduce

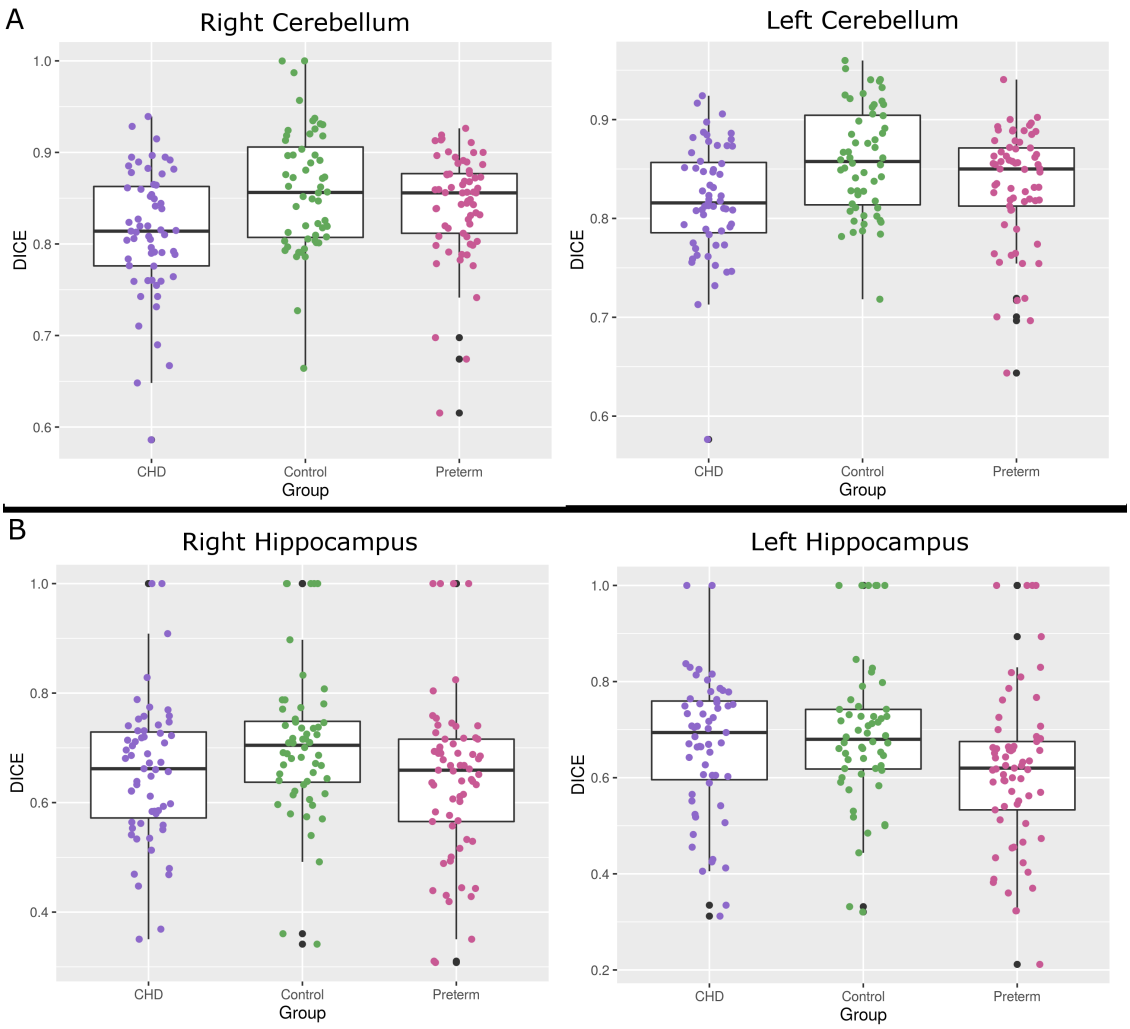

Figure 22: Comparison between raw output and manual correction. Higher Dice coefficients indicate higher similarity between substructures

	Structure			
	L Cerebellum	**R Cerebellum**	**L Hippocampus**	**R Hippocampus**
Cohort	mean (sd)	mean (sd)	mean (sd)	mean (sd)
Control	0.859 (0.05)	0.858 (0.07)	0.690 (0.15)	0.705 (0.13)
CHD	0.818 (0.06)	0.812 (0.07)	0.667 (0.14)	0.655 (0.13)
Preterm	0.833 (0.06)	0.842 (0.06)	0.615 (0.17)	0.641 (0.15)

Table 1: Comparison between raw output and manual correction. Higher Dice coefficients indicate higher similarity between substructures

the accuracy of these methods. That said, these subjects would likely fail more constrained automated segmentation methods, potentially hindering meaningful hypothesis testing, as the exclusion of these subjects would likely introduce severe selection bias.

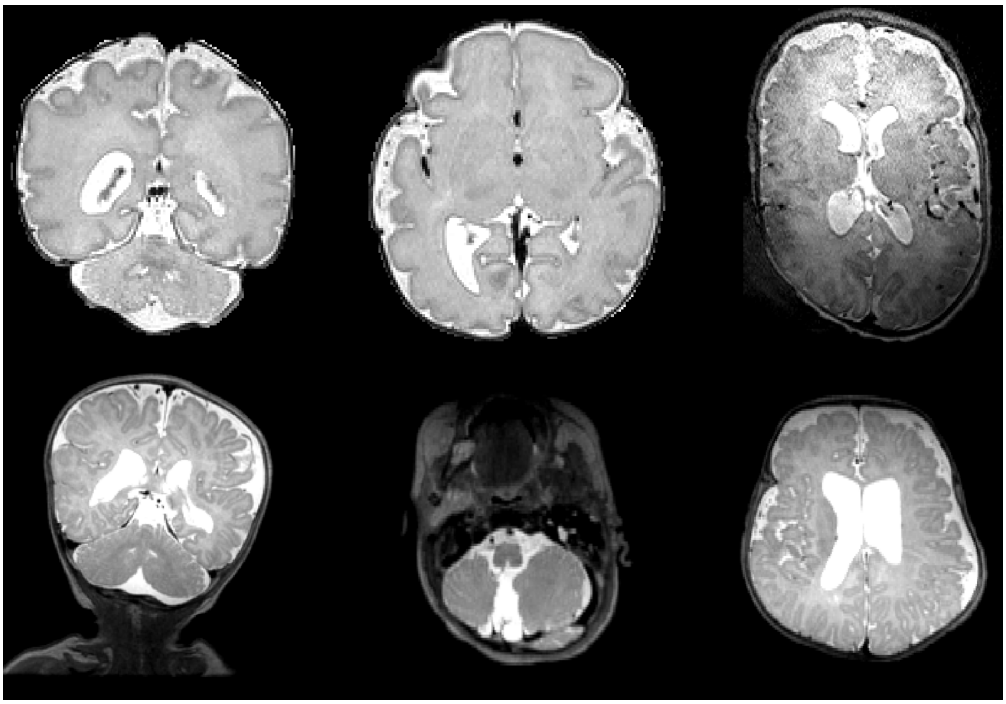

Figure 23: Sample images from the subjects with the lowest Dice coefficient across all cohorts. Note that the asymmetric and dilated ventricles pose a challenge for registration based segmentation methods. Additionally, noise and motion can reduce the accuracy of these methods. However, the exclusion of these subjects would greatly limit clinical research. These subjects would likely fail more constrained automated segmentation methods.

4.1.2 Rater Reliability

Ensuring repeatable measures in substructure segmentation is an important component of this work. Not only is this imperative for accurate volumetric analysis, the classification task is entirely dependent on an accurate delineation of the substructure's shape. Table 2 shows the mean Dice coefficient of repeated manual correction across six subjects (3 controls and 3 CHD). Two users blinded to the subject's cohort or dysplasia classification manually corrected the raw NeBSS output independently to assess the inter-rater reliability of the pipeline. For the intra-rater reliability, one user performed an additional set of corrections on the same set of subjects.

	Structure			
	L Cerebellum	R Cerebellum	L Hippocampus	R Hippocampus
Measure	mean (sd)	mean (sd)	mean (sd)	mean (sd)
Inter-Rater	0.943 (0.01)	0.951 (0.02)	0.811 (0.19)	0.84 (0.11)
Intra-Rater	0.952 (0.03)	0.941 (0.03)	0.89 (0.03)	0.87 (0.09)

Table 2: Inter- and Intra-Rater Reliability

Again, the cerebellum shows a higher inter- and intra-rater reliability compared to the hippocampus. Encouragingly, measures were very consistent across both inter- and intra-raters in the cerebellum, with the lowest mean Dice coefficient of 0.941 (+/- 0.03) between one rater measuring the right cerebellum and highest mean Dice coefficient of 0.951 (+/- 0.02) between multiple raters measuring the same substructure. The hippocampus showed somewhat diminished reproducibility, with the left hippocampus having a mean Dice coefficient of 0.811 (+/- 0.19) for inter-rater reliability, and 0.89 (+/- 0.03) for intra-rater reliability. The right hippocampus similarly showed an inter-rater reliability score of 0.84 (+/- 0.11) and intra-rater reliability of 0.87 (+/- 0.09).

4.1.3 Comparison Between 1.5 and 3.0 Tesla Field Strength

One common problem that we face when analyzing clinically acquired data, especially in retrospective studies of clinically abnormal cohorts, is that the available data is often ac-

quired using a heterogeneous set of protocols and/or magnet field strength. Clinical translational multi-center neuroprotection neonatal brain studies can be conducted at different field strengths across multiple vendor platforms, leading to heterogeneity in imaging acquisition. The latter is a large enough confounder that datasets often need to be split between field strength or outright excluded from analysis. To test the reliability of NeBSS across magnet strength, we performed a proof of concept analysis by comparing the gross structure segmentation outputs across three neonatal cohorts who had brain MRI's at our institution, using a variety of T2 weighted images acquired using both clinical and research optimized protocols on multiple vendors. The null hypothesis was that we would not observe any significant differences in volume between the cohorts when controlling for post-menstrual age (PMA) and normalizing each substructure by total brain volume (to attenuate any influences due to pathology or additional confounders). When pooling the subjects into one group, we added the subject's clinical cohort as an added covariate. Statistical significant was tested using ANOVA in R. Table 3 shows the distribution of subjects across field strengths. The cohorts are as follows: term born neonates with CHD, preterm-born neonates imaged at approximately term-equivalent age, and term born healthy controls.

	Field Strength (Tesla)	
Cohort	**1.5**	**3.0**
All Combined	17	71
CHD	7	25
Control	6	27
Preterm	4	19

Table 3: Summary of three cohorts scanned at 1.5 and 3.0 Tesla field strength

Table 4 shows the ANOVA p-values of each substructure of interest when compared between acquisitions at 1.5 and 3 Tesla field strength. No significant volumetric differences were observed in any of the substructures of interest across the cohorts. This supports the goal of NeBSS of being agnostic to both protocol and field-strength.

	Structure			
Cohort	**Cerebellum**	**CSF**	**Cortex**	**WM**
All Combined	0.605	0.998	0.450	0.957
CHD	0.119	0.692	0.795	0.242
Control	0.513	0.736	0.080	0.340
Preterm	0.731	0.805	0.765	0.194

Table 4: ANOVA p-values comparing three cohorts scanned at 1.5 and 3.0 Tesla field strength.

4.2 CLINICAL TRANSLATIONAL APPLICATION

Below is a brief description of this segmentation pipeline used to compare cerebellar volumes between a term born cohort of healthy neonates with a term born cohort of CHD neonates. This work will be submitted for publication in the upcoming months. The PMA adjusted substructure volumes are summarized in table 5 and shown in figure 24.

The CHD cohort demonstrates significantly reduced bilateral PMA adjusted cerebellar volume compared to the preterm and healthy control group ($p < 0.0000$). Similarly, we see a marginally significant decreased left hippocampus PMA adjusted volume in neonates with CHD ($p < 0.0130$), with right hippocampus showing a non-significant decrease in PMA adjusted volume ($p < 0.0520$). Surprisingly, we also see a markedly decrease in PMA adjusted deep grey matter ($p < 0.0000$), as calculated from branch B of the NeBSS pipeline. The deep grey matter probabilistic map of branch B is an aggregated mask that includes key DGM substructures including the thalamus, basal ganglia (putamen, caudate nucleus, globus palidus), and the hippocampus. We should note that the volume of this region can be influenced by enlarged ventricles, which are common in the more injured CHD neonates.

	Corrected for PCA						
	Control			CHD			
Structure	N	Mean	SD	N	Mean	SD	p <
Brainstem	29	222.6737	861.0584	79	-81.7410	1308.1393	0.2480
Cortex	29	-9351.0800	31868.1100	79	3432.6750	38885.8300	0.1160
Deep Grey Matter	29	3019.6130	4552.4130	79	-1108.4660	5041.3060	0.0000
White Matter	29	465.0342	22465.1000	79	-170.7088	30665.8900	0.9190
Infratentorial CSF	29	522.6103	4819.8440	78	-194.3038	5891.7090	0.5590
Supratentorial CSF	29	-14274.7000	21850.2900	78	5307.2600	62815.1500	0.1040
Intraventricular CSF	29	-987.2350	3379.1670	78	367.0489	6068.0750	0.2590
Whole Brain	29	14206.8770	47155.9700	78	-5282.0440	70888.3100	0.1740
Right Cerebellum	32	2556.2412	4048.6520	86	-951.1595	2841.0260	0.0000
Left Cerebellum	32	2645.2169	3881.9350	86	-984.2667	2750.6430	0.0000
Right Hippocampus	32	49.8837	233.7080	80	-19.9535	137.5246	0.0520
Left Hippocampus	32	51.2675	176.6058	81	-20.2538	116.9164	0.0130

Table 5: CHD vs. Control Volumes Corrected for PMA

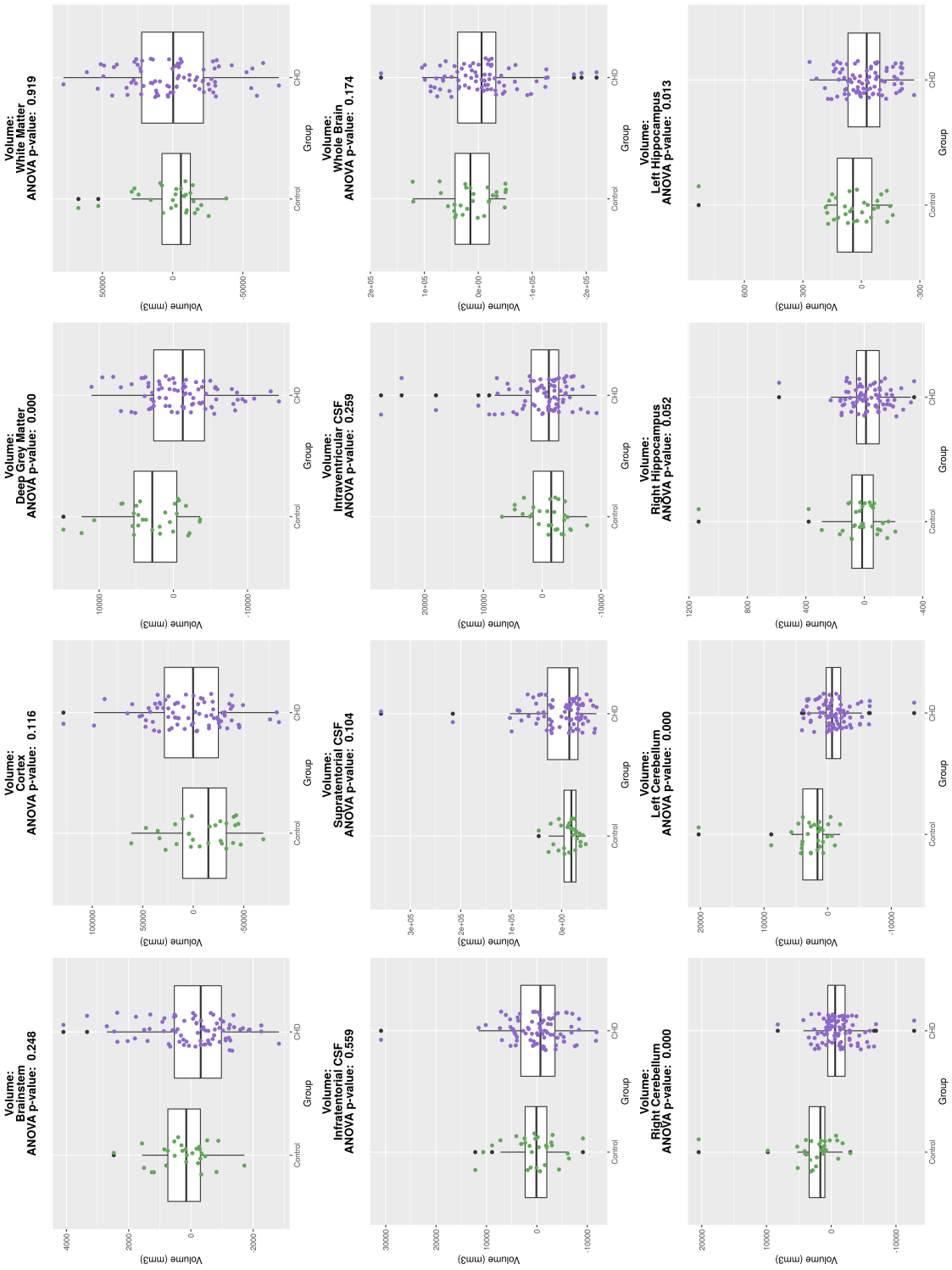

Figure 24: CHD vs. Control Volumes Corrected for PMA

5.0 CONVOLUTIONAL NEURAL NETWORKS METHODS

5.1 OVERVIEW AND CLINICAL APPLICATION

Convolutional Neural Networks are exceptionally powerful learning algorithms that have found applications in nearly every domain of machine learning. Their use in medical imaging has only begun to take off, but already they have proven useful in both classification and data mining tasks. Here, I introduce an implementation of 3-D CNNs capable of learning features from neuroimaging data. As a case study, we will use the features extracted using NeBSS as the input to train a model to predict structural dysplasia in key substructures identified as high risk for dysmaturation in neonates with congenital heart disease. However, this framework is developed to be generalizable beyond this domain. Furthermore, I provide a method of exploring the features generated by the hidden layers of the CNN, as a way of learning distinct patterns within our input data that contribute to the final classification task.

In this chapter, I will first give an in-depth description of the computational methods necessary to implement a working convolutional neural network. I will then provide an overview of the software implementation and provide a strategy for evaluating the end results.

5.2 COMPUTATIONAL METHODS

5.2.1 Gradient Descent

The following section will summarize and use the mathematical notation published by Michael A. Nielsen in Neural Networks and Deep Learning [49].

Gradient Descent (GD) is an optimization algorithm designed to minimize a given cost function by iteratively making small changes to its parameters in the opposite direction of its gradient. The gradient of a function is equivalent to its slope, generalized to higher dimensions, and indicates, given a set of values, the direction in which the function is increasing at those values. The general algorithm works in the following manner:

- Initialize random weights
- Feed first batch through
- Calculate gradient of cost function (given current results)
- Change weights by subtracting gradient*learning rate
- Feed next batch until convergence

In stochastic gradient descent, we need a cost function that must satisfy the condition of being an average

$$C = \frac{1}{n} \sum_{x} C_x \tag{5.1}$$

over cost functions C_x, where x is each training example. This property is important, as it allows us to calculate the gradient of the cost function for each training example independently, allowing us to incrementally calculate the gradient on a subset of the data (mini batches). Additionally, the cost function must only depend on the output activation of the last layer in the network. This property will become evident in the next section, where we describe how we use the backpropagation algorithm to calculate the gradient over the entire network. This gives us the starting point to calculate the gradient and backpropagate the errors.

We will use the Mean Square Error cost function as our example function:

$$C(w,b) = \frac{1}{2n}\sum_x \|y(x) - a\|^2 \tag{5.2}$$

where w,b are the weights and bias, y(x) is vector of correct labels, and a is the vector of predicted outputs. Gradient Descent minimizes the cost function by measuring a change in the function C with respect to the changes in the function parameters:

$$\Delta C \approx \frac{\partial C}{\partial b}\Delta b + \frac{\partial C}{\partial w_1}\Delta w_1 + \frac{\partial C}{\partial w_2}\Delta w_2 + ... \tag{5.3}$$

Where $b, w_1, w_2, ...$ are any arbitrary number of parameters for our network.

Using matrix notation

$$\Delta C \approx \nabla C \cdot \Delta V \tag{5.4}$$

Where ΔV is the vector of changes in the function parameters, and the gradient of the loss function with respect to each bias b and weight w_{ij} in the network is a vector of partial derivatives:

$$\nabla C \equiv (\frac{\partial C}{\partial b}, \frac{\partial C}{\partial w_1}, \frac{\partial C}{\partial w_2})^T \tag{5.5}$$

Now, the key to gradient descent is to guarantee that at every update of the algorithm, we move towards the global (or sometimes local) minimum of the function. This can be achieved simply by setting:

$$\Delta V = -\eta \nabla C \tag{5.6}$$

where η is the learning rate. We set the learning rate to a small enough value to allow us to move towards the global minimum at an adequate pace, but without overshooting it or being thrown too off course by outliers in the training data.

This gives us:

$$\Delta C \approx -\eta \nabla C \cdot \nabla C = -\eta \|\nabla C\|^2 \tag{5.7}$$

Which guarantees that $\Delta C \leq 0$, so our parameters will always decrease. This leads us to the simple update rule of Gradient Descent:

$$V \to V' = V - \eta \nabla C \tag{5.8}$$

At each iteration we change our parameters by the gradient scaled by the learning rate. In terms of the weights and bias of our network, the update rule becomes:

$$\begin{aligned} w_k \to w'_k &= w_k - \eta \frac{\partial C}{\partial w_k} \\ b_l \to b'_l &= b_l - \eta \frac{\partial C}{\partial b_l} \end{aligned} \tag{5.9}$$

Using MSE (equation 5.2), for example, averaging over each vector (sample) can be very computationally demanding. So instead, we use stochastic gradient descent (SGE). As we stated earlier, one requirement for gradient descent is that the cost function can be an average of the cost of each training example. We leverage this property by then randomly choosing a smaller subset of the entire training set, called a mini batch, under the assumption that each mini batch is proportional to using the entire dataset:

$$\sum_{j=1}^{m} \nabla C x_j \approx \sum_{x} \nabla C x = \nabla C \tag{5.10}$$

So at each iteration of gradient descent, we move our function parameters based on the gradient calculated on the mini batch, eventually using all training samples (called an epoch). Using SGE with mini batches, the weight and bias updates (equation 5.9) become:

$$\begin{aligned} w_k \to w'_k &= w_k - \frac{\eta}{m} \sum \frac{\partial C x_j}{\partial w_k} \\ b_l \to b'_l &= b_l - \frac{\eta}{m} \sum \frac{\partial C x_j}{\partial b_l} \end{aligned} \tag{5.11}$$

Note that we scale the learning rate by the size of the mini-batch to prevent outliers from skewing the direction of the descent.

5.2.2 Momentum SGD

While standard SGD is guaranteed to find a (local) minimum eventually, given a small enough learning rate, it can be rather slow. Additionally, in "ravines," or regions in which one dimension is decreasing more rapidly than the rest, SGD tends to oscillate around the local minimum. One simple improvement to SGD that attenuates this behavior is the momentum SGD method. Momentum builds on the intuition behind SGD, where we are following the "slope" of the cost function down towards its minimum point, by introducing an additional "momentum" parameter. Thus, introducing the momentum parameter to equation 5.9:

$$\begin{aligned} V_w &= \mu * V_{wt-1} - \eta \frac{\partial C}{\partial w_k} \\ w_k \rightarrow w'_k &= w_k - V_w \\ V_b &= \mu * V_{bt-1} - \eta \frac{\partial C}{\partial b_l} \\ b_l \rightarrow b'_l &= b_l - V_b \end{aligned} \tag{5.12}$$

The momentum parameter μ acts to accelerate the parameter vector towards any direction that has a consistent gradient. Note that we introduce a velocity vector V which has "memory," in the sense that it is set as the previous iteration's gradient direction. This allows it to accelerate the gradient descent in the direction in which is was last increasing, and decelerate it in dimensions in which the direction is changing. The V parameter has the same dimension as the weights and bias, and is initialized at 0. As we iterate through the data, it picks up momentum towards the function's minima. The parameter μ acts as a scaling factor on the speed in which V accelerates. This is an additional hyperparameter that needs to be optimized, and is generally initialized at 0.5 and increased to 0.9 - 0.95 after learning has stabilized.

5.2.3 Regularization

Overtraining is always a concern with highly complex machine learning algorithms. Overtraining occurs when the model implicitly learns features specific to the training data, effectively memorizing it, but does not generalize to external datasets. To reduce these effects,

we can impose restrictions on our cost function and hidden layers. One method of controlling overtraining is L2 regularization. L2 regularization adds an additional term to the cost function

$$\frac{\lambda}{2n}\sum_{w} w^2 \tag{5.13}$$

where λ is an additional hyperparameter, n is the number of subjects in the training batch and w are the weights. This method adds a penalty for higher weight values, attenuating runaway effects that can lead to overfitting the data.

5.2.4 Layer Drop-Out

A complimentary method of preventing overfitting is layer drop-out. Layer drop-out is a method of ensuring more generalizability in classification function. Layer drop-out randomly removes a pre-determined number of neurons from the final fully-connected layers at each training iteration. This prevents the learning algorithm from relying too heavily on any one neuron, enforcing more generalization distributed across the entire network space instead of localized neurons.

5.2.5 Hidden Layer Visualization

A significant secondary result from this work are the features extracted from the neural networks. As described in the previous sections, neural networks create abstractions of the original input image, effectively reducing the dimensionality of the data and creating sets of features that are useful to the given classification task. These intermediate filters (the activations of the hidden layers, projected as feature maps) can be informative, giving us insight into what set of values in the original dataset are most important for accurate classification, generating a lower dimensionality representation of the input data. In the context of structural dysplasia, these features may indicate which sub-regions or particular shapes are most informative to the classification task. This could potentially provide insight into the biological and structural features that contribute to a given substructure's malformation.

These can be used to objectively refine the existing set of clinical guidelines for diagnosing these abnormalities, as the current method is more reliant on qualitative observation and clinical experience. Of course, neural networks do not learn with the explicit intention of generating informative or human comprehensible sets of features. The parameters learned in the hidden layers of a neural network are traditionally treated as a black box. We have no direct control on the features learned at each layer, and it has been shown that each individual unit does not often hold any meaningful semantic information, but rather the combination of features within the entire space hold this higher level of abstraction[76]. Therefore the hidden layers are not guaranteed to be useful in this context, but still warrant investigation. We can leverage the highly structured nature of our dataset, as each hidden layer's feature map will encode the local features within the input that contribute to the final classification. Because each subject's substructure is linearly registered into a common space, we can view the mean layer activations for each group as a proxy for the anatomical features that inform the final classification function.

Figure 25 shows synthetic examples of possible filters that can be learned by a CNN. Randomly distributed filters (A) may still result in good classification accuracy, but provide no anatomically interpretable benefits. Random filters may also be symptomatic of an overfitting algorithm. Geometric filters (B) are often seen in image classification algorithms with no structural coherence of the input data. These filters are typically found in earlier layers of deep networks, and are thought to be used in further layer abstractions to combine to form more complex features. While mechanistically interpretable, they also do not provide useful anatomical insight in this use case. Localized filters (C), where only a subsection of the input image activates the filter, can be useful for identifying specific regions of the input image that highly contribute to the final classification. While potentially desirable due to their specificity, localized filters are not likely to be observed when classifying complex structural morphology, but would theoretically be useful in lesion classification or other general discrete feature detection. Structurally coherent activation maps (D) retain the structure of the input data, and selectively activate regions of the input dataset that contribute to the final classifier. In this work, we attempt to leverage this property to identify complex patterns within the input structure that ultimately differentiate normal and dysplastic subtypes.

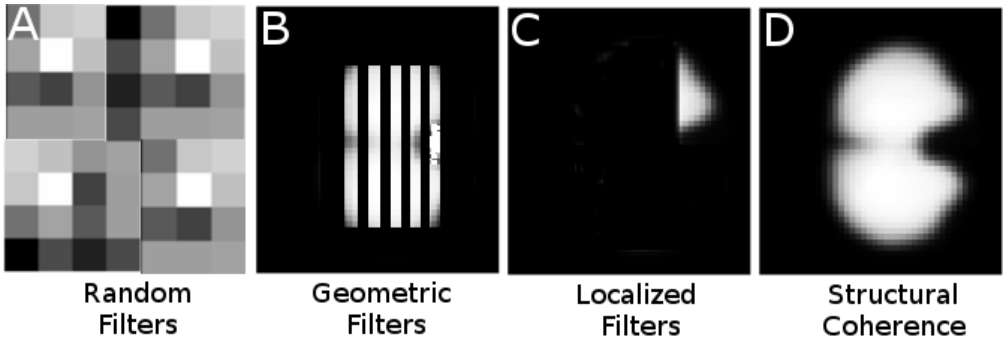

Figure 25: Possible learned filter outcomes when training CNNs. The filters learned by a CNN are not guaranteed to be human interpretable, even when achieving good classification accuracy, as the final classification layer uses a linear combination of all available filters. However, in this work we structure our input data onto a common space, and enforce structural coherence of the learned filters.

5.3 MODEL CREATION AND TRAINING

5.3.1 Adjusting The Hyperparameters

Designing an effective neural network model can be an exercise in patience and luck. There are no absolute guidelines as to what architectures work best for certain problems. We can, however, follow certain heuristics to improve our chances of landing on a model that works to our satisfaction. We can start by fixing as many of the hyperparameters as we can by examining previously successful implementations and modeling our parameters after them.

Based on existing successful implementation CNN's, starting with a final softmax activation function in conjunction with the negative log likelihood cost function is a sensible approach. The remaining layers can initially use the rectified linear unit (ReLU) activation function, as it greatly reduces learning slowdown and more directly conveys the strength of activation at each neuron. This can be especially useful when decoding the abstractions of the hidden layers. Initial runs should begin with simple models before adding to the complexity. Therefore regularization and drop-out measures should be avoided until the

network at least demonstrates an ability to learn from the data. An additionally useful tool in conjunction with ReLU, once an acceptable framework is established, is L2 regularization. L2 regularization adds an additional term to the cost function that penalizes large weights. This helps to keep our parameters from increasing out of control, under the principle that a more parsimonious set of weights is more likely to generalize well. Should our CNN perform poorly in the more complex classification task, this is the logical next step.

This leaves us with the number of hidden layers, number of feature maps per layer, size of the local receptive field, and the learning rate. We can search through possible values for each of these parameters by using a combination of grid and random search. By setting a range of possible values for each hyperparameter, we can randomly sample from within this range and iterate over the possible combinations of parameters, recording which models perform best. For the number of hidden layers, my strategy will first be to start with a relatively simple model, such as the single hidden layer model shown in figure 9, and increase the complexity until we achieve acceptable results. We can iteratively increase the number of layers, followed by the number of feature maps per layer, and so on. The size of the local receptive field will change depending on the size of the structure we are trying to classify. Initially setting a LRF size of 10% of the total size of the bounding box surrounding the substructure of interest in each dimension can yield good results. The intuition in this use case is that the features that are most useful to the classifier will be around the edges of the substructures. This will reduce the number of parameters we have to estimate, especially in the center of the substructures where we don't expect to find useful features.

5.3.2 Dataset

90 term-born neonates with congenital heart disease and 40 term-born healthy controls were prospectively recruited at our institution in an Institutional Review Board approved study with parent consent. Infants were scanned at close to term equivalent age, or when infants were deemed clinically stable. Mean gestational age was 38.0 weeks (+/-2.9) in the infants with CHD, and 41.2 weeks (+/- 3.8) in the control group. Mean post-menstrual age (PMA) at time of scan was 42.4 (+/-6.9) and 43.5 (+/- 5.5) weeks for children with CHD and con-

trols, respectively. Infants were scanned on a 3T Siemens Skyra MRI (Siemens, Erlangen, Germany) without sedation using a 32-channel head coil. 18 infants in the control group were scanned on a 3T GE HDXT. Only infants who completed the volumetric imaging portion of the protocol were included in this study, and a cut-off of 52 weeks post-menstrual age was used to control for age-related morphological changes. Infant T2 images were processed through the NeBSS pipeline described in chapter 3, and the input to the classification algorithm was the binarized structure linearly registered onto the template space.

We have previously described a pattern of dysmaturation in a subset of this population[39]. Each neonate's MRI was reviewed by an expert neuroradiologist blinded to their clinical history and classified as normal or dysplastic following existing qualitative imaging criteria developed by Barkovich, et. al [77]. For cerebellar classification, although the cerebellar vermis and cerebellar lobes were scored independently, we considered the presence of dysplasia in either bilateral substructures as a dysplastic cerebellum. For the hippocampus, we considered the presence of dysplasia and/or malrotation to indicate dysplasia. Because of this, we used bilateral hippocampus as the input. Infants identified to have hypoplastic substructures, but no evidence of structural dysplasia were not classified as abnormal in this study. 17 (18.9%) infants with CHD were diagnosed with a dysplastic cerebellum. The total incidence rate in the entire dataset was 13.1%. 34 infants (37.8%) with CHD were diagnosed with a dysplastic hippocampus, giving a total incidence rate in the entire dataset of 26.2%.

An additional challenge in CNNs applied to medical imaging is that there is often a very low incidence of abnormal cases in the training dataset. This can significantly slow down training, or impede it altogether, as the algorithm sees a disproportionate amount of normal cases through each iteration. To remedy this, we bootstrap the data by randomly sampling from the set of abnormal cases and artificially inflating the dataset to contain a more proportional ratio of controls to abnormal cases. The cerebellum was bootstrapped to have a final training dataset containing 47% dysplastic substructures, and the hippocampus had a final proportion of 68% abnormal cases. To attenuate the effects of overtraining the algorithm by oversampling the same small subset of cases, we introduce a small amount of random translation (3-5 voxels) in 3 directions to each case in the dataset. This helps prevent the introduction of fixed-position based artifacts and aids in generalization of the

parameters.

5.3.3 Evaluation

To validate our classification results, we performed a 10-fold cross validation. This is done by partitioning the dataset into 10 independent sets, and at each iteration using 9 sets as the training set and the remaining data for validation of the classification accuracy. To ensure that the training and validation sets remain independent, the resampled subjects from the bootstrapping method described in the previous section remained within their own validation sets without cross-over into the remaining dataset. We trained each validation run for a total of 100 epochs. The best performing parameters from the validation runs were then used in a final learning run of 700 epochs to generate the activation maps described in the next section.

5.4 SOFTWARE IMPLEMENTATION

We developed a 3D CNN built using Python and the Theano framework[58, 49], named CRBNet, with customized routines for the 3D convolution and neuroimage processing. The neuroimage processing component of the pipeline leverages the Neuroimaging in Python (Nipy) library[72]. The logical choice of computational framework for this implementation was Theano. It is a well established and extensible framework with excellent documentation and native to Python. While TensorFlow is tempting due to its strong corporate backing, it is still in its relative infancy, lacking in features and optimization compared to Theano. It would be trivial, however, to refactor the relevant codebase to migrate to TensorFlow should it prove superior to Theano in future iterations. The full code can be found at https://www.github.com/PIRCImagingTools/CRBNet. It is free, open source, and licensed under the BSD License.

CRBNet takes as input a configuration file in JSON format that contains all the necessary parameters and inputs of the network. This configuration file is used to construct

the computational graph and initialize the training. Appendix B shows a sample JSON control file used as input for the network. As the network learns, it automatically generates diagnostics metrics in real time until completion. Figure 26 shows an example diagnostic image generated by CRBNet. These are useful monitoring tools that inform the user of the current performance. If it is obvious that the algorithm has stopped learning from the data, it may be useful to terminate early and re-initialize with new parameters.

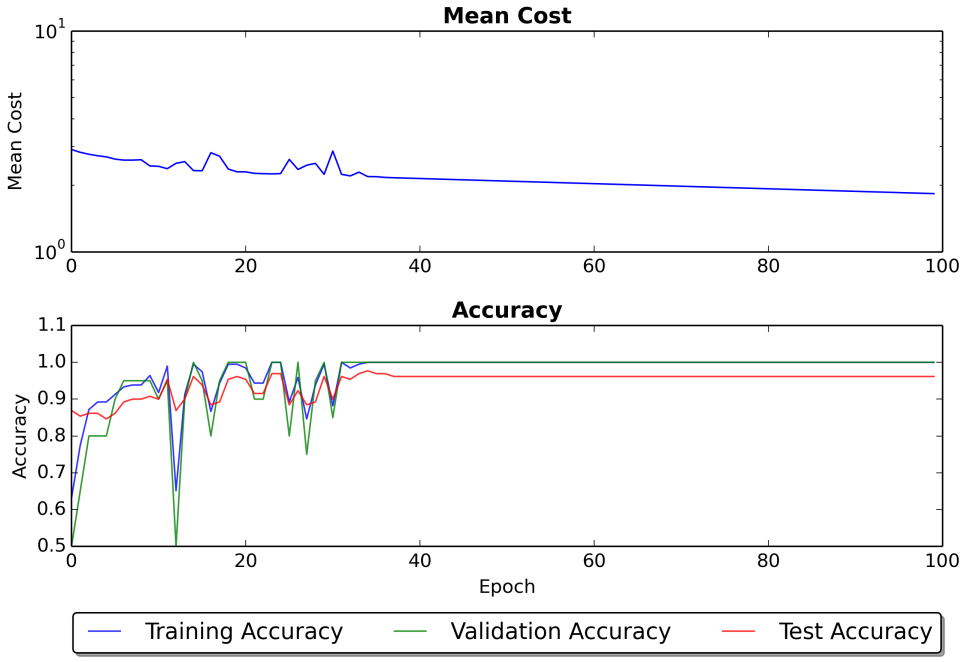

Figure 26: Sample network metrics output in real time

6.0 CONVOLUTIONAL NEURAL NETWORKS RESULTS

6.1 ESTIMATING CNN HYPERPARAMETERS

6.1.1 Initial Hyperparameters

In order to test our implementation of convolutional neural networks using a 3-dimensional input, we first created a simple synthetic dataset on which we can expect a high rate of success. We selected two randomly chosen neonatal cerebelli, extracted using NeBSS, and generated an inflated dataset by adding small amounts (5-20 voxels in each direction) of random translations to each cerebellum. The final dataset consisted of 50 cerebelli generated from each subject, for a total of 100 input images. The classification task was simply to identify the original subject the image was generated from. We achieved 100% classification accuracy using a combination of convolution layers and fully connected layers, summarized in table 6. The final hyperparameters chosen were the rectified linear units activation function for every layer, a learning rate of 0.03, and mini-batch size of 10 (due to hardware limitations). The stopping criteria was set to 40 epochs, however the network achieved 100% classification in 28 epochs.

This simplified classification task allows us to test the implementation of the CNN's in a controlled environment. The features necessary to simply distinguish two different cerebelli are naturally much simpler than those necessary for detecting subtle dysplasia across multiple cohorts.

Layer	Layer Type	Input Dimension	# of Neurons / Feature Maps	LRF	Pooling
1	Convolution/Max Pool	117 x 159 x 126	10	5 x 5 x 5	2 x 2 x 2
2	Convolution/Max Pool	56 x 77 x 61 x 10	6	5 x 5 x 5	2 x 2 x 2
3	Fully Connected	157,248	200	-	-
4	Fully Connected	200	100	-	-
5	Softmax	100	2	-	-

Table 6: Summary of layers in a successful 3D Convolutional Neural Network for classifying neonatal cerebelli

6.1.2 Complexity vs. Resources

The compute resources available can place severe constraints on the scale of the neural network. Convolution neural networks can quickly exceed a standard workstation's available resources as the number of parameters and layers increases, especially when the input is 3-dimensional. We can take advantage of the convolutional architecture, however, and leverage the power of layer abstractions and max-pooling as you move down the network.

This can be modeled as a balance between a compute footprint and a memory footprint. The compute footprint is the number of parameters your network must update at every iteration. Fully connected layers in general have exponentially more parameters to calculate, which takes additional compute time at each iteration. However, the number of parameters has relatively little effect on the memory requirements. The memory footprint comes from the amount of storage your network will take up in memory while learning. This has a fixed cost based on the size of the input data, and a variable cost based on the network parameters. In fully connected networks, this scales linearly with the number of neurons in each layer. In convolutional networks, however, it becomes a function of the number of feature maps and max-pooling steps at each layer. Since a feature map is a structural representation of the input data, without a max-pooling step, you effectively multiply the size of your original input by the number of feature maps. This can become unwieldy if care is not taken in designing the architecture. Table 7 shows the memory and compute footprints for varying

network architectures. We can use this as a basis for choosing the optimal parameters to begin training the network.

Total Layers	Architecture	Memory Footprint	Compute Footprint
2	100 Full, Softmax	630,102	63,000,200
2	200 Full, Softmax	630,202	126,000,400
2	300 Full, Softmax	630,302	189,000,600
2	400 Full, Softmax	630,402	252,000,800
3	5 Conv, 100 Full, Softmax	970,662	34,056,520
3	10 Conv, 100 Full, Softmax	1,311,222	68,112,840
3	15 Conv, 100 Full, Softmax	1,651,782	102,169,160
3	20 Conv, 100 Full, Softmax	1,992,342	136,225,480
3	25 Conv, 100 Full, Softmax	2,332,902	170,281,800
4	25 Conv, 25 Conv, 100 Full, Softmax	2,497,902	16,541,800
5	25 Conv, 25 Conv, 25 Conv, 100 Full, Softmax	2,508,702	1,161,800
6	25 Conv, 25 Conv, 25 Conv, 25 Conv, 100 Full, Softmax	2,508,852	136,800
6	40 Conv, 30 Conv, 20 Conv, 10 Conv, 100 Full, Softmax	3,561,282	136,760
6	10 Conv, 20 Conv, 30 Conv, 40 Conv, 100 Full, Softmax	1,456,422	152,840

Table 7: Compute and memory footprints for varying architectures of a network with initial input dimension of 100 x 90 x 70. Here we assume a 2 x 2 x 2 max pooling step following each layer

On this principle, the general heuristic becomes using fewer feature maps at the early layers, and increasing them as you move down. The power of CNNs can be seen from this simple experiment. We create lower dimensional representations of our input data while still building a powerful classifier - all with decreased computational and memory footprints.

6.1.3 Final CNN Parameters

Table 8 shows the final parameters for our chosen architecture for classifying dysplastic cerebelli. It consists of a total of 7 hidden layers, with 4 initial convolutional layers, followed by 2 fully connected layers and a final softmax classification layer. Each convolutional layer is followed by a max-pooling procedure. Max pooling is a method of down sampling the image by taking the maximum value within the given filter. This has the advantage of decreasing the search space at each subsequent layer and de-noising the data at each step. The initial learning rate (η) was set to 0.005 with a scheduled rate decay of 0.5 * η every 40 epochs. We used the negative log-likelihood cost function, with an added L2 regularization

hyperparameter (λ) set to 0.01. We used a layer dropout parameter of 0.3. Finally, we implemented a momentum update method with an initial value of 0.5, increased to 0.9 after a stabilization period of 15 epochs. Modern computer vision CNNs have achieved excellent results and improved learning speed using the Rectified Linear Units function (equation 2.5) as the activation function. However, we achieved rather mediocre results with ReLU in our application. This is likely a result of the sparse nature of the inputs, which result in exploding and/or vanishing gradients during training. Instead, our final architecture uses the tanh function (equation 2.4), which has traditionally performed well in many classification tasks, at the cost of slower learning at the saturation extrema compared to newer activation functions such as ReLU.

Layer	Layer Type	Input Dimension	# Neurons/ Feature Maps	LRF	Pooling
1	Convolutional/Max Pool	100 x 90 x 70	10	4 x 4 x 4	2 x 2 x2
2	Convolutional/Max Pool	48 x 43 x 33 x 10	15	2 x 2 x 2	2 x 2 x2
3	Convolutional/Max Pool	23 x 21 x 16 x 15	25	2 x 2 x 2	2 x 2 x2
4	Convolutional/Max Pool	11 x 10 x 7 x 25	50	2 x 2 x 2	2 x 2 x2
5	Fully Connected	5 x 4 x 3 x 50 = 3,000	300	-	-
6	Fully Connected	300	100	-	-
7	Softmax	100	2	-	-

Table 8: Final CNN Parameters chosen for classifying dysplastic cerebelli

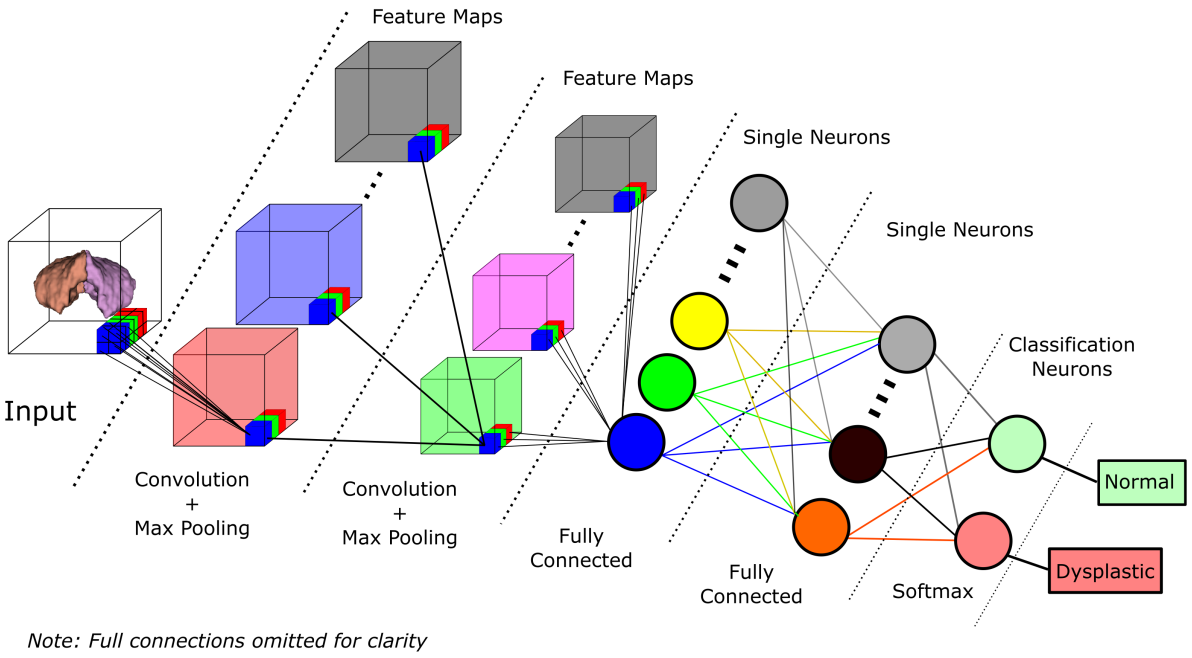

Figure 27: Simplified overview of CRBNet. The Framework is flexible, allowing for any configuration of layers and classifiers

6.2 CROSS VALIDATION

6.2.1 Cerebellum Results

Figure 28 shows the mean cost at each epoch the 10 cross-validation runs in the cerebellum. While some variance is expected due to the random initialization of weights, all runs converge within 50 epochs. Figure 29 shows the classification accuracy for each run. The training set is a bootstrapped dataset with an inflated incidence of abnormal cases. The data is partitioned into 10 independent sets, where at each run 9 are used to train the network and the final set is used as the validation set. The test set is the original dataset (without the bootstrapped cases added in). All runs achieved 100% classification accuracy in the training set within 50 epochs. The average classification accuracy on the validation set was 0.985 +/- 0.0241, with several reaching 100% classification accuracy.

6.2.2 Hippocampus Results

The same methods used to for hyperparameter estimation and training in the cerebellum dataset were used in the hippocampal data. Additional attempts to increase the complexity of the network were made after the chosen cerebellar parameters proved unsuccessful. We performed a grid search of hyperparameters with increasing complexity ranging from 7 to 9 total hidden layers, and number of feature maps ranging from 5 to 50 at each hidden layer. However, no set of hyperparameters was able to achieve a higher classification accuracy of 66.7%. The current limiting factor for further increasing the complexity is the memory capabilities of the computing system. To visualize the added complexity of the hippocampus compared to the cerebellum, we created spatial distribution maps for each substructure, shown in figure 30. The spatial distribution maps show the proportion of subjects in the dataset that had their substructure morphology present at each voxel within the standard template space.

We see much tighter spatial cohesion in the cerebellum, a relatively simple substructure, when compared to the hippocampus. The hippocampus shows much higher variance at its border, indicating much higher structural variance in the dataset.

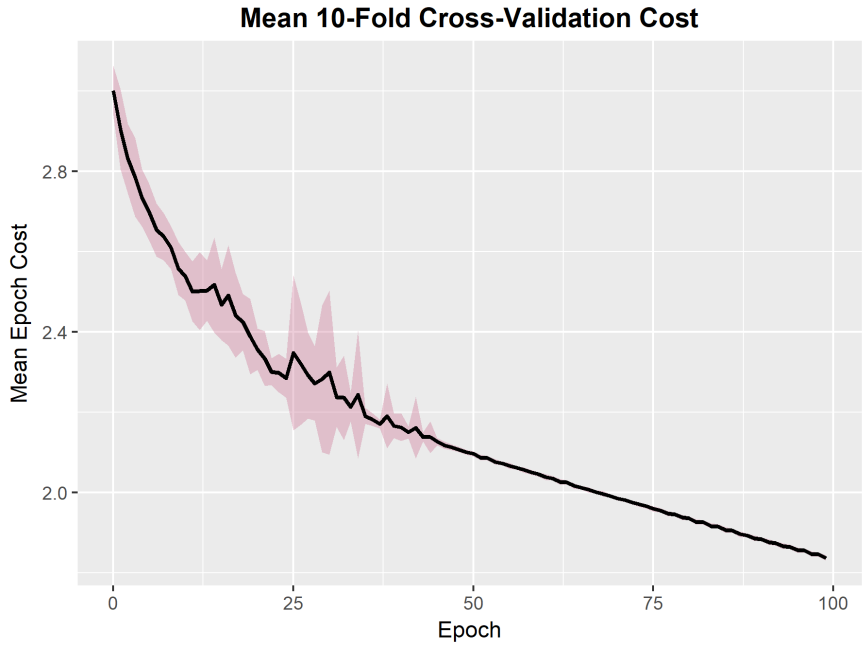

Figure 28: 10-Fold Cross validation mean and standard deviation cost across all cerebellum runs. Cost function was the negative log-likelihood function with an L2 regularization parameter of 0.01. While some variance is expected due to the random initialization of weights, all runs converge within 50 epochs.

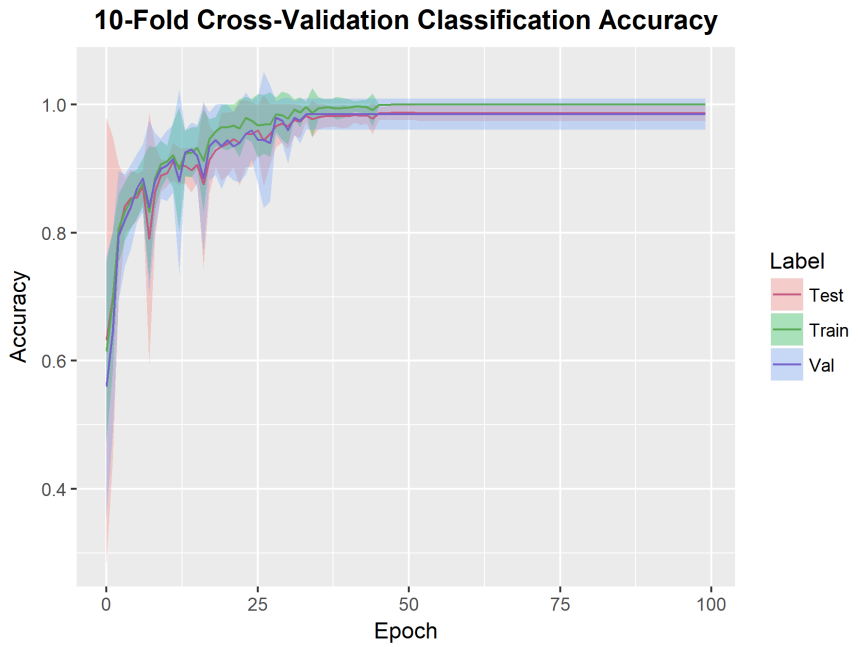

Figure 29: 10-Fold Cross validation mean and standard deviation classification error for each dataset. The training set is a bootstrapped dataset with an inflated incidence of abnormal cases. The data is partitioned into 10 independent sets, where at each run 9 are used to train the network and the final set is used as the validation set. All runs achieved 100% classification accuracy in the test set within 50 epochs

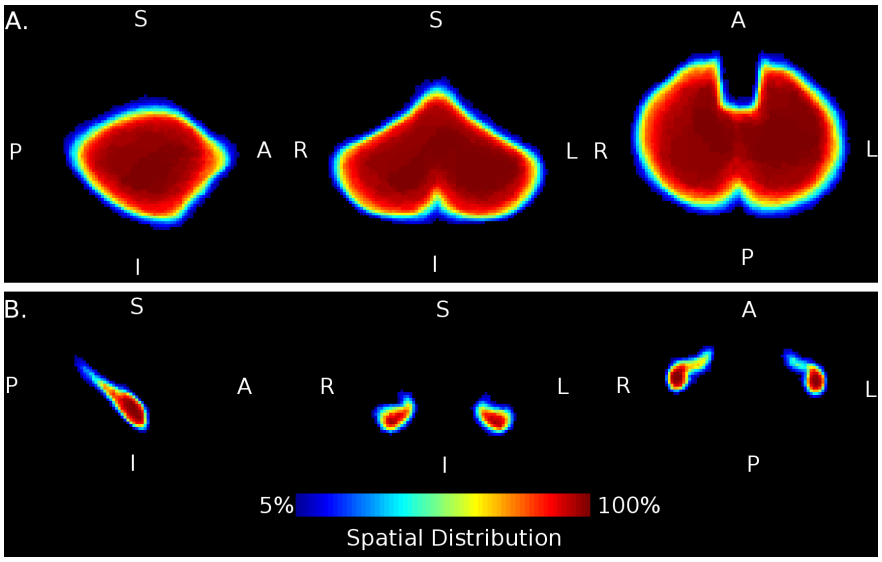

Figure 30: Spatial Distribution Maps of A) Cerebellum and B) Hippocampus. The spatial distribution maps show the proportion of subjects in the dataset that had their substructure morphology present at each voxel within the standard template space. Note the significantly higher spatial coherence of the cerebellum compared to the hippocampus. Anatomical labels indicate Anterior-Posterior, Superior-Inferior, and Left-Right axes.

6.3 HIDDEN LAYER VISUALIZATION

The first layer's mean activations for the entire dataset are shown in Figure 31. Intensities are scaled to show the contrast in activation range in each filter. We see that each filter distinctly delineates the cerebellum, with some filters showing higher activations limited to the perimeters of the substructure, serving as potential edge detectors. Comparatively, the activations in layer 2 (Figure 32) show more dramatic delineations of the peripheral substructure. Note that filters with visually similar activations were removed from the image for clarity. Subsequent layers show similar increase in the range of activations within each filter, however, as they are sub-sampled at each hidden layer, they become less interpretable. Figure 33 shows the activations for the third layer.

We can then look at the difference in activations between the dysplastic and normal cohorts. The difference map of the first layer is shown in Figure 34. Blue regions are areas in which we see higher activation in the dysplastic cerebelli, and red is increased in the normal substructures. Each filter seems to show a predilection for primarily classifying either normal or dysplastic substructures, however, some overlap is observed. We see a clear pattern where the normal substructures show more defined delineation of the cerebellar lobes while the dysplastic substructures show more defined posterior cerebellar regions. This contrast becomes even more pronounced in the second layer's difference maps (Figure 35). Additionally, regions corresponding to the cerebellar vermis show a strong discriminatory power, seen as strongly differentiated regions of activation in the superior, center regions of the feature maps. As with the mean activation maps, subsequent layers lose anatomic interpretability due to subsampling, but show a stronger discriminatory pattern 36.

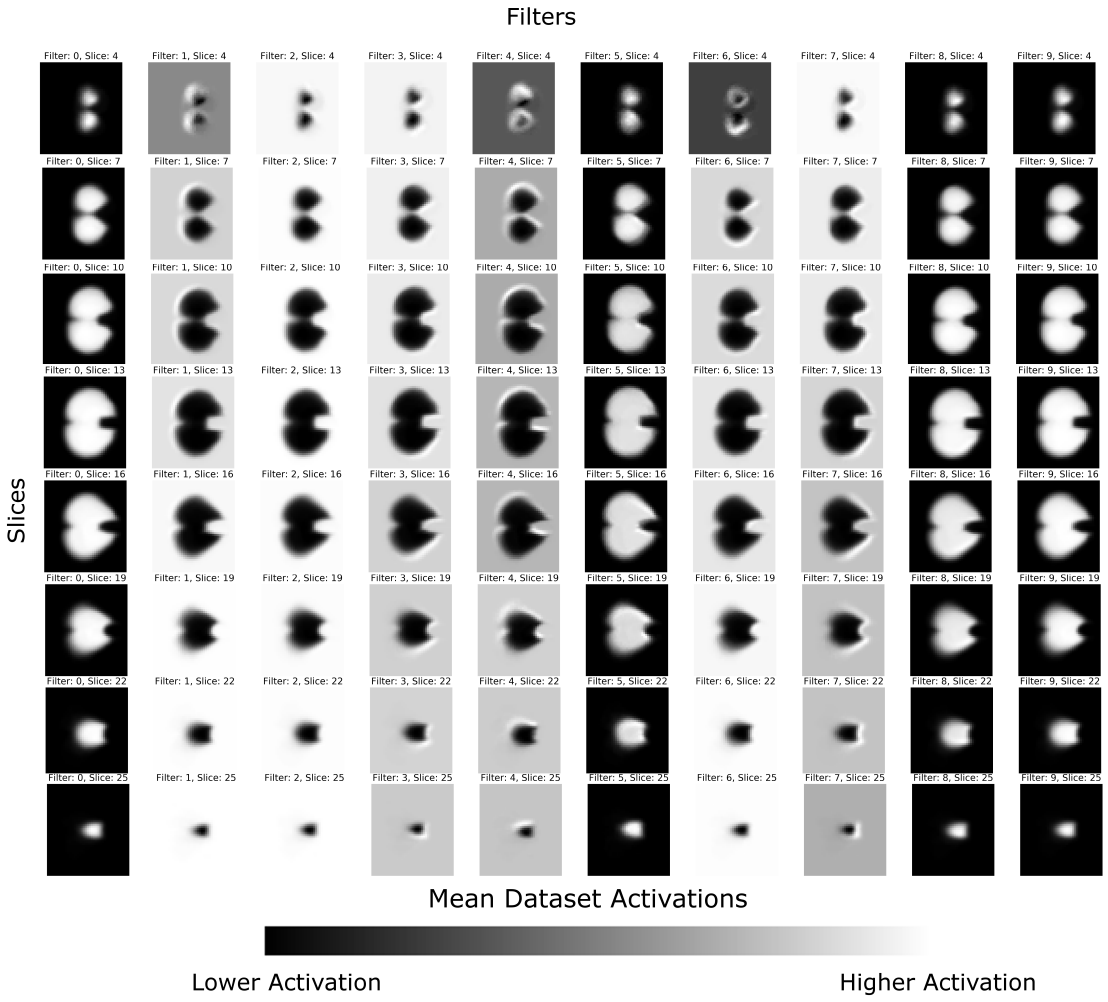

Figure 31: First convolutional layer mean activations. The first layer acts as an edge detector and general classifier for gross cerebellar substructure.

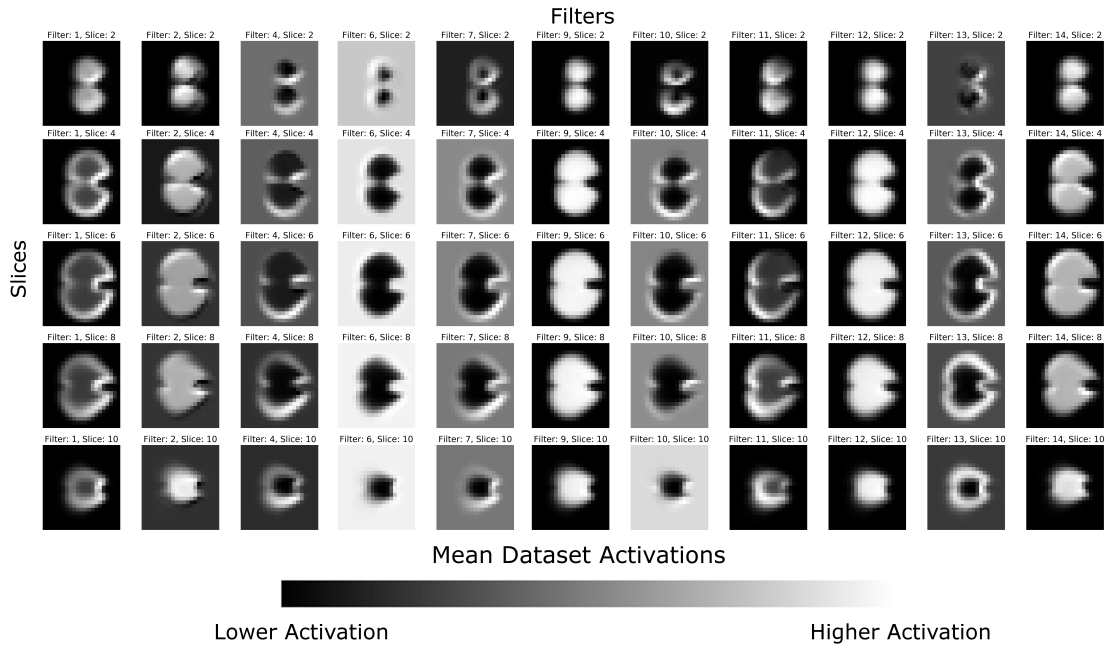

Figure 32: Second convolutional layer mean activation. Note that visually similar filters were removed for clarity. Compared to the first layer activations, we see more discriminant delineations of cerebellar lobes, vermis, and posterior fossa.

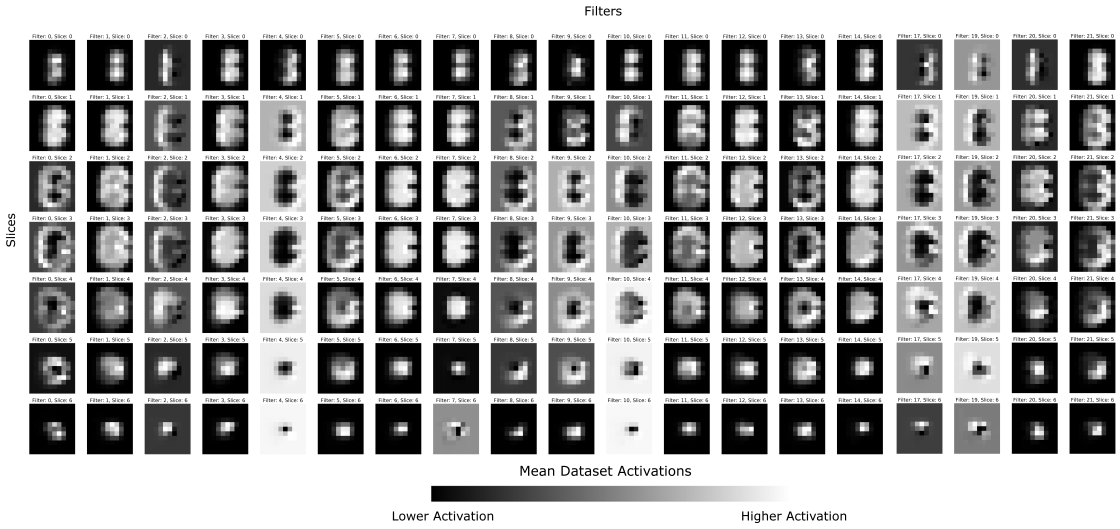

Figure 33: Third Layer Activations

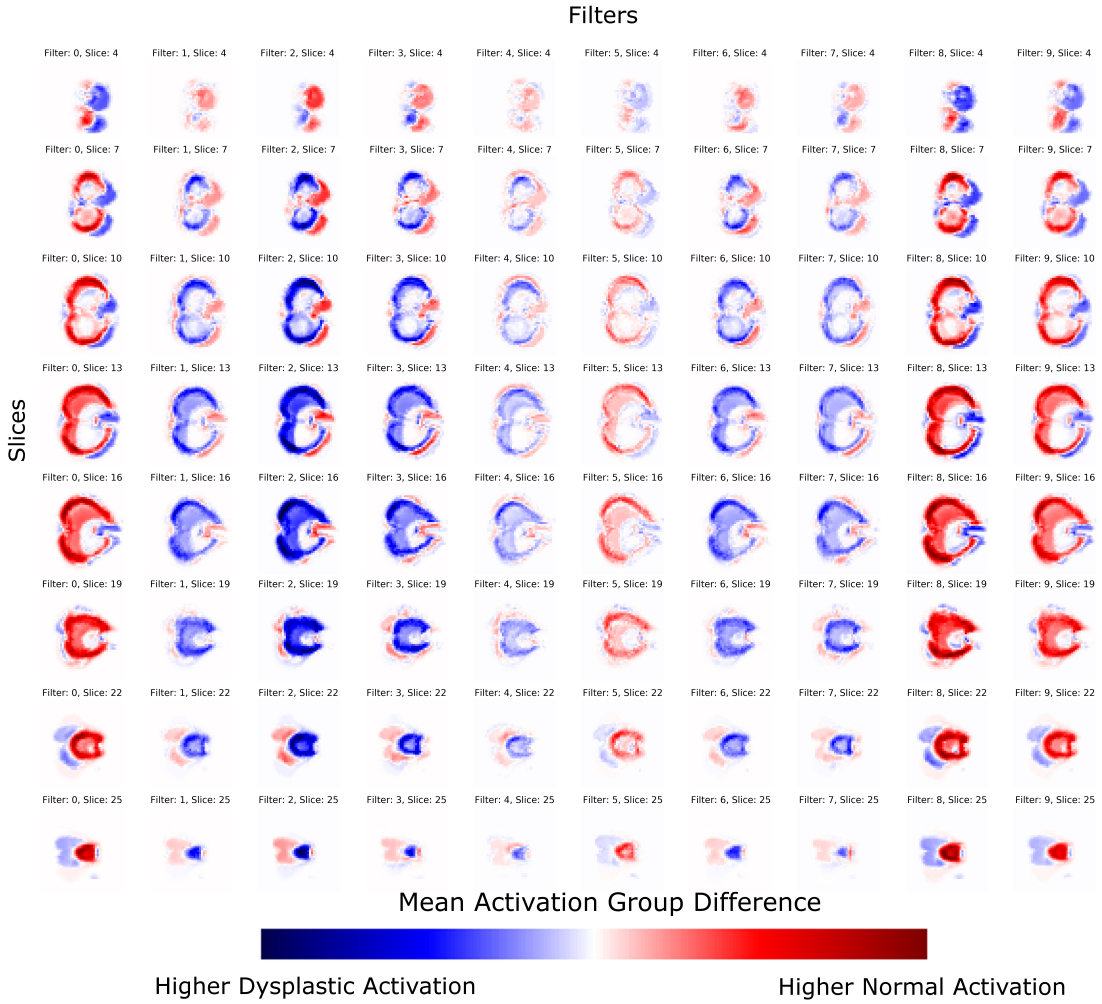

Figure 34: First convolutional layer activation difference maps between control and dysplastic cerebelli.

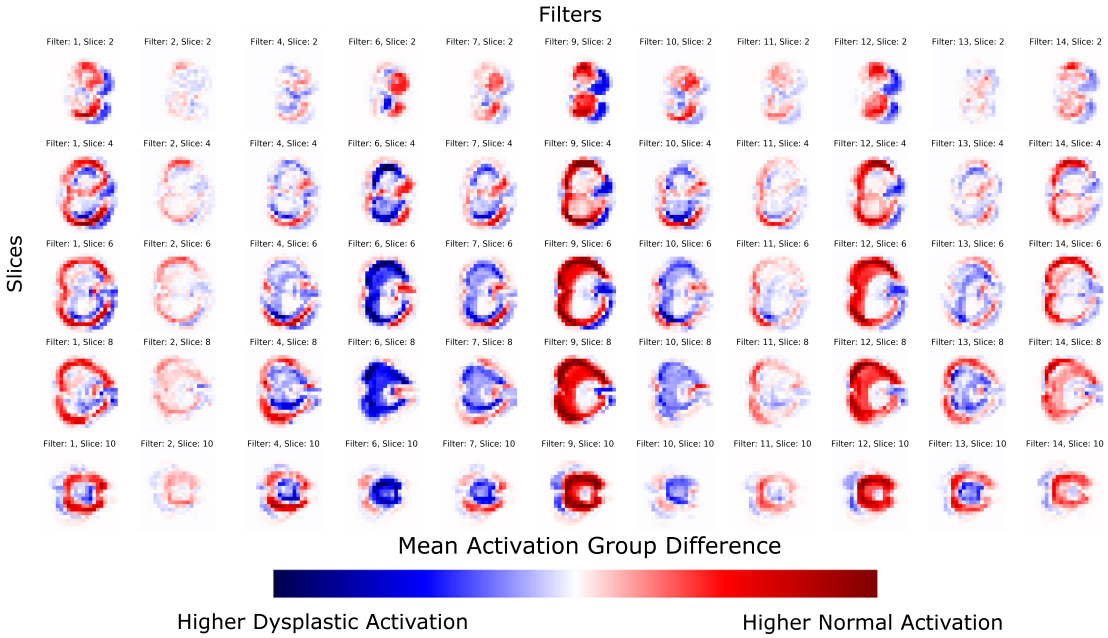

Figure 35: Second convolutional layer activation difference between control and dysplastic cerebelli. More robust discriminatory features are observed between normal and dysplastic cerebelli.

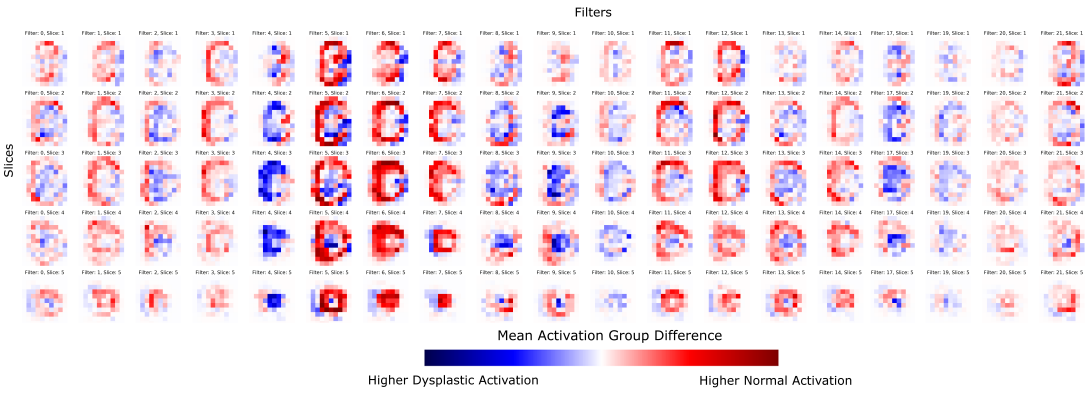

Figure 36: Third convolutional layer activation difference between control and dysplastic cerebelli. More robust discriminatory features are observed between normal and dysplastic cerebelli, but with markedly decreased resolution.

7.0 DISCUSSION

7.1 NEBSS

We have introduced NeBSS, a robust semi-automated neonatal brain segmentation pipeline. NeBSS fills a need in clinical translational research in neonatal imaging, where existing automated or semi-automated implementations are too rigid to be successfully applied to clinically significant cohorts. NeBSS is able to reliably parcellate neonatal images acquired using heterogeneous imaging protocols and magnet field strength, with minimal manual correction, especially when compared to traditional manual segmentation methods. The output volumes show no observable bias as a result of magnet field strength, and we are confident that it can be applied to heterogeneous clinical data. We also show that the final manual correction step is robust and reproducible, with high inter- and intra-rater reliability. Finally, we apply this method to perform volumetric analysis of a clinical dataset of neonates born with CHD. The software is further strengthened by being fully open source, allowing the research community to incrementally improve on its performance and usability transparently. The implementation is still in its infancy, with room for improvement in both computational efficiency and usability.

This work is not without limitations, however. NeBSS shows lower than desired Dice similarity coefficients between the raw output and the final manually corrected substructures. Manual correction can still be a laborious task, and the goal of this work is to eliminate as much of it as possible. Of course, manual correction is still an improvement over the traditional method of manual substructure delineation from scratch. Future work will focus on reducing the computational time required as well as increasing the accuracy of segmentation of poor quality images. This can be achieved by optimizing different segments of the pipeline.

First, we can iteratively create a study specific neonatal template from existing subjects that have been processed through NeBSS and manually corrected. Having a study specific template, especially when dealing with patients with ventriculomegaly, can significantly increase registration accuracy. Having an additional option of an injured brain template to serve as the input atlas can ultimately improve performance in challenging cohorts beyond patients with CHD. Additionally, improving the performance of ANTS will significantly decrease the processing time of the registration steps. Refactoring the code to run the ALBERT template registrations in parallel, utilizing multiple cores, will reduce the compute time by a factor of 4. Finally, creating a pre-segmentation scheduling procedure that heuristically estimates the optimal ANTS parameters prior to initializing the algorithm may reduce compute time by reducing the amount of iterations needed for the algorithm to converge. Combining this scheduling procedure with the selection of optimal cohort specific prior templates as discussed above would greatly improve both computational time and registration accuracy of the pipeline, while decreasing the required amount of user intervention throughout the process.

7.2 CRBNET

We have introduced a computational framework for the application of 3D Convolutional Neural Networks on a limited clinical dataset, showing excellent performance using cross-validation. Modern deep learning methods are increasingly being used in very sophisticated classification problems involving highly dimensional data. However, they often require vast amounts of training examples to successfully learn an adequate classifier. In this work, we overcome this challenge by first extracting the substructures of interest and registering them into a common space. This greatly reduces the computational search space and increases the observed effect size, creating a relatively trivial application of deep learning as the final step. Although we have implemented a supervised learning method of classification, we can look at the activation maps generated by the algorithm as data driven learned features that provide insight into the underlying morphology of cerebellar dysplasia. Our observations correspond

with recent work in fetal MRI which observed shape abnormalities in patients diagnosed with CHD[78], specifically in cerebellar vermis. By and large, studies in infants with CHD focus on volumetric alterations. Current guidelines rely on qualitative observations that lack the sensitivity for more subtle morphological malformations that may have a clinical impact in long term development. By more clearly defining key regions of the cerebellum which are vulnerable to dysmaturation, we can more aptly create objective guidelines. The cerebellum undergoes rapid development late in the gestation period, at which time a disrupted catch up growth period may lead to irreparable long term damage. The primary function of the cerebellum has classically been associated with motor learning, tone, coordination, and language. However, more recent development indicates cerebellar modulation of higher cognitive functions, including working memory, processing speed, and executive functioning[79]. Furthermore, aberrant cerebellar development has been proposed to influence social and affect regulation, a term labeled cerebellar cognitive affective syndrome (CCAS)[80]. These deficits have long term consequences, and early intervention is imperative. Developing more sensitive methods for early detection of cerebellar impairment is an important step in the palliative care of infants born with congenital heart disease.

The results obtained in the hippocampal classification task are disappointing but not surprising. The hippocampus is a much more structurally complex region compared to the cerebellum, in addition to being significantly smaller. These two factors combine for a much more challenging classification task, as evidenced by our negative results. It is likely that our limited dataset does not model enough of the complexity necessary to discriminate this dysplastic substructure. One possible additional contribution to this is that we chose to include both individual dysplasia and malrotation as our predictor. The intuition is that a CNN would be able to detect a relative position abnormality given bilateral hippocampi, however, it is likely that the low incidence of malrotation in the dataset contributed more noise than signal to the classifier.

This study has several limitations. First, our sample size is small. The incidence of cerebellar dysplasia in this population is low, as there is a clear selection bias against more injured infants who may not have been deemed healthy enough to enroll in the study or have additional injuries that would exclude them, including gross ventriculomegaly and intra-

ventricular hemorrhage. Additionally, we validate our results using cross-validation. While this helps prevent overtraining of the algorithm, it does not completely eliminate biases implicit to the dataset, such as population-specific variance and data acquisition parameters. However, our framework's ability to discriminate subtle dysplastic cerebelli is encouraging. We chose to train a 3D CNN, rather than use transfer learning of existing high-performing models such as AlexNet, to preserve the 3D structure of the input substructures. While using an existing feature rich model trained on a much larger dataset could result in more powerful discriminatory performance, it would require pre-processing of the input data that would diminish the interpretability of the structural morphology associated with the learned classifiers. Therefore, we believe that generating interpretable features is a worthwhile trade-off. Finally, the nature of the structure dysplasia scores is a limitation. While these scores are qualitative measures assigned by a trained radiologist, there are no rigorous criteria by which to follow. It is a metric driven by experience, which can be subject to certain biases. However, our ability to consistently detect structural dysplasia in the cerebellum supports the reproducibility of this diagnostic measure.

Our future work is geared towards applying this method on an independently acquired dataset for external validation, and improving its performance on more complex morphology. This framework is designed to be generalizable to any structural classification task in neuroimaging, and we aim to broaden its use by leveraging existing large datasets and computational frameworks. Currently, the code is developed to run on a single local workstation. Expanding this to be easily scalable to a super computer or cloud environment would greatly broaden the capabilities of both computation and dataset sizes CRBNet can handle.

8.0 CONCLUSION

In this thesis we present a computational framework for the extraction and classification of dysplastic substructures in neonatal MRI. We achieved excellent classification accuracy using 10-fold cross-validation. Furthermore, the hidden layer activations provide insight into the morphological characteristics of the patterns of dysmaturation of brain substructures in a neonatal population at risk of long term cognitive deficits. By modeling the problem using training data scored by an expert neonatal radiologist, we can use the learned parameters as a proxy for the intuitive features used by clinicians to diagnose these subtle patterns. This can be used to develop more objective clinical diagnostic guidelines. Finding biomarkers of dysmaturation during the neonatal period is an important but challenging task. Aberrant structural and functional markers are often too subtle to detect at an individual level, and their cognitive effects can only be detected later in the child's life, at which point interventions may have limited efficacy. Therefore, identifying potential problems early is imperative for the development and implementation of early interventions.

The long-term direction of this work is towards incorporating this framework into a clinical decision support and automated diagnosis systems. However, the limited availability of clinical data makes it infeasible to develop intelligent systems that detect global level anomalies. Therefore creating an ensemble of specialized systems that work in parallel to inform clinicians of a wide range of potential markers of dysmaturation is a more realistic approach. Having this expert knowledge readily available to clinicians natively in an electronic health system has the potential to greatly improve quality of care and treatment planning. It is imperative that these built-in systems provide pertinent information without overloading clinicians with irrelevant or unreliable information. To achieve this, further work is needed to improve the accuracy and computation time of the substructure extraction. To make it

a truly effective tool, it is necessary to remove the manual correction step altogether. It is our hope that this can be accomplished with access to larger datasets and further iterations of the computational methods used here.

More generally, in this work we have shown the feasibility of using highly sophisticated, cutting edge machine learning methods in a domain with relatively limited availability of data. The advent of "Big Data Science" has greatly broadened the scope and ambitions of machine learning applications. More and more benchmark datasets are being released, with training sets containing thousands of samples and labels, providing a rich environment for pushing the boundaries of computation and algorithms. And while true that testing new methods is better suited using large benchmark datasets, making the leap into practical, real-world applications should be the long term goal. Transfer learning is emerging as the next logical step into accomplishing this. However, as we have touched upon in this work, transfer learning has the potential to sacrifice specificity within the domain for the sake of high classification accuracy. The inferential insights gained from tailoring a learning algorithm specifically to the given domain, in this case retaining the relative 3-dimensional structure of the input brain substructures, may be equally or more beneficial to biomedical research. The specific patterns of dysmaturation learned in this work may inform future research into the genetic and mechanistic processes that contribute to the observed phenotype. Ultimately, by understanding the limitations of both the computational methods as well as the underlying biology, we can constrain our data into a more manageable search space.

Finally, the code developed for this project is open source and published under the BSD License. Although we are validating this work on the specific setting of dysmaturation in neonates, our overarching goal is to provide a generalized framework to give future researchers the tools necessary to implement deep neural networks using any structural dataset. We believe this to be a significant contribution to both translational medicine and informatics.

APPENDIX A

COMPUTING THE GRADIENT USING BACKPROPAGATION

The following chapter will summarize and use the mathematical notation published by Michael A. Nielsen in Neural Networks and Deep Learning [49]. Here we will explain how we calculate the gradient for our network using the backpropagation algorithm. We will use the following notation for each weight in our network: w^l_{jk}, where l is the current layer, j is the neuron of the l layer, and k is the neuron of the previous layer (l-1). So the output of any give neuron becomes:

$$a^l_j = \sigma(\sum_k w^l_{jk} a^{l-1}_k + b^l_j) \tag{A.1}$$

Where σ is the chosen activation function. We can simplify this by using matrix notation:

$$\boldsymbol{a}^l = \sigma(\boldsymbol{W}^l \boldsymbol{a}^{l-1} + \boldsymbol{b}^l) \tag{A.2}$$

The backprop algorithm works under the following assumptions:

1) The cost function can be written as an average over individual cost functions:

$$C = \frac{1}{n} \sum_x C_x \tag{A.3}$$

i.e. the MSE cost of a single example is:

$$C_x = \frac{1}{2}\|y - a^L\|^2 \tag{A.4}$$

This allows the backprop algorithm to compute the cost and $\frac{\partial C_x}{\partial w}$, $\frac{\partial C_x}{\partial c}$ for a single example, then average all examples in a batch.

2) C can be written as a function of the outputs from the network.

This allows us to calculate the gradient by first adding a small error term Δz_j^l to a neuron's *weighted output* (before applying the cost function), and the neuron will output $\sigma(z_j^e + \Delta z_j^e)$. Thus, the overall cost changes by $\frac{\partial C}{\partial z_j}\Delta z_j^l$. Here we can define $\delta_j^l \equiv \frac{\partial C}{\partial z_j^l} \equiv$ the error of neuron j in layer l in terms of **z** (pre-activation), not **a** (post-activation).

With this principle, we derive 4 equations which allow us to calculate the gradient of any layer in our network:

1) Error in the output layer, δ^L

$$\delta_j^L = \frac{\partial C}{\partial a_j^L}\,\sigma'(Z_j^L) \tag{A.5}$$

Which, in the case of the MSE cost function, can be broken down down as:

$$\frac{\partial C}{\partial a_j^L} \equiv C' \equiv (a_j - y_j) \tag{A.6}$$

i.e. how fast cost changes as a function of this neuron, and $\sigma'(Z_j^L) \equiv$ how fast the activation function changes at Z.

Equation A.5 in matrix form becomes:

$$\begin{aligned} \delta^L &= \nabla_a C \odot \sigma'(Z^L) \\ \delta^L &= (\boldsymbol{a}^L - \boldsymbol{y}) \odot \sigma'(Z^L) \end{aligned} \tag{A.7}$$

Where $\odot$ is the Hadamard product $\equiv s \odot t \equiv$ the element wise multiplication of vectors:

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} \odot \begin{bmatrix} 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 \cdot 3 \\ 2 \cdot 4 \end{bmatrix} = \begin{bmatrix} 3 \\ 8 \end{bmatrix}$$

2) Layer error δ^l in terms of the error in the next layer δ^{l+1}

$$\delta^l = ((w^{l+1})^T \delta^{l+1}) \odot \sigma'(Z^l) \tag{A.8}$$

This propagates the error back through the layers. First we use equation A.7 to calculate δ^L, which is the previous layer, then continue moving backwards using Z. Now that we know how to calculate the error, we can calculate the rate of change of the cost as we change the weights and bias for each neuron at each layer using the following two equations:

3) Rate of change of the cost wrt any bias in the network

$$\frac{\partial C}{\partial b^l_j} = \delta^l_j \tag{A.9}$$

The error δ^l_j is exactly equal to the rate of change.

4) Rate of change of the cost wrt any weight in the network

$$\frac{\partial C}{\partial w^l_{jk}} = a^{l-1}_k \ \delta^l_j \equiv a_{in}\delta_{out} \tag{A.10}$$

As we can see, at each neuron we use the layer error (propagated from the next layer) and the post-activation output of the previous layer to calculate the rate of change.

APPENDIX B

SAMPLE JSON NETWORK SETTINGS

CRBNet takes as input a configuration file in Javascript Object Notation (JSON) format that contains all the necessary parameters and inputs of the network. This configuration file is used to construct the computational graph and initialize the training. Below is a sample JSON control file used as input for the network. The user specifies the datasets and text labels used for training, validation, and testing. The global network parameters are the minibatch size, number of epochs to run, the learning rate (eta), the optional parameter of the proportion by which to lower the learning rate (etadecay), if eta decay is used, the user also must specify the interval (in epochs) at which the decay rate is applied. Overfitting can be controlled using the regularization parameter (lmbda), and p dropout, which is the proportion of neurons which are randomly omitted at each epoch (globally applied to fully connected layers unless specified). The optimization method can be specified by the "descent method" parameter, with options of "SGD" (stochastic gradient descent) or "momentum." If momentum is chosen, the "mu" hyperparameter must also be specified. Finally, the input dimensions (of each training sample) must be specified, and if existing parameters are to be used at the start of learning, then the "restart" flag can be chosen.

Each layer must then be specified. Convolutional layers must be given the number of feature maps, size of the local receptive field ("lrf"), shape of the max pooling filter ("max pool"), and the activation function to be used. Options of activation function are "Sigmoid," "Tanh," "ReLU" (Rectified Linear Units), and "lReLU" (leaky ReLU). Fully connected layers receive the number of neurons in the layer ("n out"), a layer-specific dropout parameter

(optional), and the activation function. The final softmax layers receives as parameter the number of classes in the final output ("n out").

```
{
 "Data":
  {"TRAIN_STACK" : "/path/to/training/data.nii.gz",
   "TRAIN_LABELS" : "/path/to/training/Labels.txt",
    "VALID_STACK" :  "/path/to/validation/data.nii.gz",
    "VALID_LABELS" : "/path/to/validation/Labels.txt",
    "TEST_STACK" :  "/path/to/test/data.nii.gz",
    "TEST_LABELS" :  "/path/to/test/labels.txt"},
 "mini_batch_size" : 5,
 "epochs" : 700,
 "eta": 0.005,
 "eta_decay": 0.5,
 "eta_interval": 40,
 "lmbda": 0.01,
 "p_dropout" : 0.3,
 "descent_method" : "momentum",
 "mu" : 0.5,
 "input_dims" : [1, 100, 90, 70],
 "restart" : false,
 "Structure":
  {
   "layer_0":
    {"type": "Conv",
     "feature_maps": 10,
     "lrf": [4, 4, 4],
     "max_pool" : [2, 2, 2],
     "activation_function": "Tanh"},
   "layer_1":
    {"type": "Conv",
     "feature_maps": 15,
     "lrf": [2, 2, 2],
     "max_pool" : [2, 2, 2],
     "activation_function": "Tanh"},
   "layer_2":
    {"type": "Conv",
     "feature_maps": 25,
     "lrf": [2, 2, 2],
     "max_pool" : [2, 2, 2],
     "activation_function": "Tanh"},
   "layer_3":
    {"type": "Conv",
     "feature_maps": 50,
     "lrf": [2, 2, 2],
     "max_pool" : [2, 2, 2],
```

```
      "activation_function": "Tanh"},
    "layer_4":
     {"type": "Full",
      "n_out": 300,
      "p_dropout" : 0.0,
      "activation_function": "Tanh"},
    "layer_5":
     {"type": "Full",
      "n_out": 100,
      "p_dropout" : 0.0,
      "activation_function": "Tanh"},
    "layer_6":
     {"type": "Softmax",
      "n_out": 2,}
    }
}
```

APPENDIX C

RELEVANT WORK IN NEONATAL DEVELOPMENT

The work presented in this thesis was built on a foundation of years of research in neonatal development. The following chapter gives a brief overview of published works I have been a part of in this topic. While some extend beyond the scope of this thesis, they frame a larger picture in the setting of dysmaturation in both preterm birth and congenital heart disease.

C.1 ABNORMAL MICROSTRUCTURE IN NEONATES

Abnormal cerebral microstructure in premature neonates with congenital heart disease.

Paquette LB, Wisnowski JL, **Ceschin R**, Pruetz JD, Detterich JA, Del Castillo S, Nagasunder AC, Kim R, Painter MJ, Gilles FH, Nelson MD, Williams RG, Bluml S, Panigrahy A.

AJNR Am J Neuroradiol. 2013 Oct;34(10):2026-33. doi: 10.3174/ajnr.A3528. Epub 2013 May 23.

Abnormal cerebral microstructure has been documented in term neonates with congenital heart disease, portending risk for injury and poor neurodevelopmental outcome. Our hypothesis was that preterm neonates with congenital heart disease would demonstrate diffuse cerebral microstructural abnormalities when compared with critically ill neonates without congenital heart disease. A secondary aim was to identify any association between

microstructural abnormalities, white matter injury (eg, punctate white matter lesions), and other clinical variables, including heart lesions. With the use of tract-based spatial statistics, an unbiased, voxelwise method for analyzing diffusion tensor imaging data, we compared 21 preterm neonates with congenital heart disease with 2 cohorts of neonates without congenital heart disease: 28 term and 27 preterm neonates, identified from the same neonatal intensive care unit. Compared with term neonates without congenital heart disease, preterm neonates with congenital heart disease had microstructural abnormalities in widespread regions of the central white matter. However, 42% of the preterm neonates with congenital heart disease had punctate white matter lesions. When neonates with punctate white matter lesions were excluded, microstructural abnormalities remained only in the splenium. Preterm neonates with congenital heart disease had similar microstructure to preterm neonates without congenital heart disease. Diffuse microstructural abnormalities were observed in preterm neonates with congenital heart disease, strongly associated with punctate white matter lesions. Independently, regional vulnerability of the splenium, a structure associated with visual spatial function, was observed in all preterm neonates with congenital heart disease.

Regional vulnerability of longitudinal cortical association connectivity: Associated with structural network topology alterations in preterm children with cerebral palsy.

Ceschin R, Lee VK, Schmithorst V, Panigrahy A.

Neuroimage Clin. 2015 Sep 6;9:322-37. doi: 10.1016/j.nicl.2015.08.021. eCollection 2015.

Preterm born children with spastic diplegia type of cerebral palsy and white matter injury or periventricular leukomalacia (PVL), are known to have motor, visual and cognitive impairments. Most diffusion tensor imaging (DTI) studies performed in this group have demonstrated widespread abnormalities using averaged deterministic tractography and voxel-based DTI measurements. Little is known about structural network correlates of white matter topography and reorganization in preterm cerebral palsy, despite the availability of new therapies and the need for brain imaging biomarkers. Here, we combined novel post-processing methodology of probabilistic tractography data in this preterm cohort to improve spatial and regional delineation of longitudinal cortical association tract abnormalities us-

ing an along-tract approach, and compared these data to structural DTI cortical network topology analysis. DTI images were acquired on 16 preterm children with cerebral palsy (mean age 5.6 4) and 75 healthy controls (mean age 5.7 3.4). Despite mean tract analysis, Tract-Based Spatial Statistics (TBSS) and voxel-based morphometry (VBM) demonstrating diffusely reduced fractional anisotropy (FA) reduction in all white matter tracts, the along-tract analysis improved the detection of regional tract vulnerability. The along-tract map-structural network topology correlates revealed two associations: (1) reduced regional posterior-anterior gradient in FA of the longitudinal visual cortical association tracts (inferior fronto-occipital fasciculus, inferior longitudinal fasciculus, optic radiation, posterior thalamic radiation) correlated with reduced posterior-anterior gradient of intra-regional (nodal efficiency) metrics with relative sparing of frontal and temporal regions; and (2) reduced regional FA within frontal-thalamic-striatal white matter pathways (anterior limb/anterior thalamic radiation, superior longitudinal fasciculus and cortical spinal tract) correlated with alteration in eigenvector centrality, clustering coefficient (inter-regional) and participation co-efficient (inter-modular) alterations of frontal-striatal and fronto-limbic nodes suggesting re-organization of these pathways. Both along tract and structural topology network measurements correlated strongly with motor and visual clinical outcome scores. This study shows the value of combining along-tract analysis and structural network topology in depicting not only selective parietal occipital regional vulnerability but also reorganization of frontal-striatal and frontal-limbic pathways in preterm children with cerebral palsy. These finding also support the concept that widespread, but selective posterior-anterior neural network connectivity alterations in preterm children with cerebral palsy likely contribute to the pathogenesis of neurosensory and cognitive impairment in this group.

C.2 ABNORMAL RESTING STATE NETWORKS IN PRETERMS

Alterations of resting state networks and structural connectivity in relation to the prefrontal and anterior cingulate cortices in late prematurity.

Degnan AJ, Wisnowski JL, Choi S, **Ceschin R**, Bhushan C, Leahy RM, Corby P,

Schmithorst VJ, Panigrahy A.

Neuroreport. 2015 Jan 7;26(1):22-6. doi: 10.1097/WNR.0000000000000296.

Late preterm birth is increasingly recognized as a risk factor for cognitive and social deficits. The prefrontal cortex is particularly vulnerable to injury in late prematurity because of its protracted development and extensive cortical connections. Our study examined children born late preterm without access to advanced postnatal care to assess structural and functional connectivity related to the prefrontal cortex. Thirty-eight preadolescents [19 born late preterm (34-36 /7 weeks gestational age) and 19 at term] were recruited from a developing community in Brazil. Participants underwent neuropsychological testing. Individuals underwent three-dimensional T1-weighted, diffusion-weighted, and resting state functional MRI. Probabilistic tractography and functional connectivity analyses were carried out using unilateral seeds combining the medial prefrontal cortex and the anterior cingulate cortex. Late preterm children showed increased functional connectivity within regions of the default mode, salience, and central-executive networks from both right and left frontal cortex seeds. Decreased functional connectivity was observed within the right parahippocampal region from left frontal seeding. Probabilistic tractography showed a pattern of decreased streamlines in frontal white matter pathways and the corpus callosum, but also increased streamlines in the left orbitofrontal white matter and the right frontal white matter when seeded from the right. Late preterm children and term control children scored similarly on neuropsychological testing. Prefrontal cortical connectivity is altered in late prematurity, with hyperconnectivity observed in key resting state networks in the absence of neuropsychological deficits. Abnormal structural connectivity indicated by probabilistic tractography suggests subtle changes in white matter development, implying disruption of normal maturation during the late gestational period.

Altered Structural and Functional Connectivity in Late Preterm Preadolescence: An Anatomic Seed-Based Study of Resting State Networks Related to the Posteromedial and Lateral Parietal Cortex.

Degnan AJ, Wisnowski JL, Choi S, **Ceschin R**, Bhushan C, Leahy RM, Corby P, Schmithorst VJ, Panigrahy A.

PLoS One. 2015 Jun 22;10(6):e0130686. doi: 10.1371/journal.pone.0130686. eCollection 2015.

Late preterm birth confers increased risk of developmental delay, academic difficulties and social deficits. The late third trimester may represent a critical period of development of neural networks including the default mode network (DMN), which is essential to normal cognition. Our objective is to identify functional and structural connectivity differences in the posteromedial cortex related to late preterm birth. Thirty-eight preadolescents (ages 9-13; 19 born in the late preterm period (32 weeks gestational age) and 19 at term) without access to advanced neonatal care were recruited from a low socioeconomic status community in Brazil. Participants underwent neurocognitive testing, 3-dimensional T1-weighted imaging, diffusion-weighted imaging and resting state functional MRI (RS-fMRI). Seed-based probabilistic diffusion tractography and RS-fMRI analyses were performed using unilateral seeds within the posterior DMN (posterior cingulate cortex, precuneus) and lateral parietal DMN (superior marginal and angular gyri). Late preterm children demonstrated increased functional connectivity within the posterior default mode networks and increased anti-correlation with the central-executive network when seeded from the posteromedial cortex (PMC). Key differences were demonstrated between PMC components with increased anti-correlation with the salience network seen only with posterior cingulate cortex seeding but not with precuneus seeding. Probabilistic tractography showed increased streamlines within the right inferior longitudinal fasciculus and inferior fronto-occipital fasciculus within late preterm children while decreased intrahemispheric streamlines were also observed. No significant differences in neurocognitive testing were demonstrated between groups. Late preterm preadolescence is associated with altered functional connectivity from the PMC and lateral parietal cortex to known distributed functional cortical networks despite no significant executive neurocognitive differences. Selective increased structural connectivity was observed in the setting of decreased posterior interhemispheric connections. Future work is needed to determine if these findings represent a compensatory adaptation employing alternate neural circuitry or could reflect subtle pathology resulting in emotional processing deficits not seen with neurocognitive testing.

C.3 DEEP GREY MATTER DEVELOPMENT AND CONNECTIVITY IN PRETERMS

Reduced thalamic volume in preterm infants is associated with abnormal white matter metabolism independent of injury.

Wisnowski JL, **Ceschin RC**, Choi SY, Schmithorst VJ, Painter MJ, Nelson MD, Blml S, Panigrahy A.

Neuroradiology. 2015 May;57(5):515-25. doi: 10.1007/s00234-015-1495-7. Epub 2015 Feb 10.

Altered thalamocortical development is hypothesized to be a key substrate underlying neurodevelopmental disabilities in preterm infants. However, the pathogenesis of this abnormality is not well-understood. We combined magnetic resonance spectroscopy of the parietal white matter and morphometric analyses of the thalamus to investigate the association between white matter metabolism and thalamic volume and tested the hypothesis that thalamic volume would be associated with diminished N-acetyl-aspartate (NAA), a measure of neuronal/axonal maturation, independent of white matter injury. Data from 106 preterm infants (mean gestational age at birth: 31.0 weeks4.3; range 23-36 weeks) who underwent MR examinations under clinical indications were included in this study. Linear regression analyses demonstrated a significant association between parietal white matter NAA concentration and thalamic volume. This effect was above and beyond the effect of white matter injury and age at MRI and remained significant even when preterm infants with punctate white matter lesions (pWMLs) were excluded from the analysis. Furthermore, choline, and among the preterm infants without pWMLs, lactate concentrations were also associated with thalamic volume. Of note, the associations between NAA and choline concentration and thalamic volume remained significant even when the sample was restricted to neonates who were term-equivalent age or older. These observations provide convergent evidence of a neuroimaging phenotype characterized by widespread abnormal thalamocortical development and suggest that the pathogenesis may involve impaired axonal maturation.

Abnormal development of thalamic microstructure in premature neonates

with congenital heart disease.

Paquette LB, Votava-Smith JK, **Ceschin R**, Nagasunder AC, Jackson HA, Bluml S, Wisnowski JL, Panigrahy A.

Pediatr Cardiol. 2015 Jun;36(5):960-9. doi: 10.1007/s00246-015-1106-8. Epub 2015 Jan 22.

Preterm birth is associated with alteration in corticothalamic development, which underlies poor neurodevelopmental outcomes. Our hypothesis was that preterm neonates with CHD would demonstrate abnormal thalamic microstructure when compared to critically ill neonates without CHD. A secondary aim was to identify any association between thalamic microstructural abnormalities and perioperative clinical variables. We compared thalamic DTI measurements in 21 preterm neonates with CHD to two cohorts of neonates without CHD: 28 term and 27 preterm neonates, identified from the same neonatal intensive care unit. Comparison was made with three other selected white matter regions using ROI manual-based measurements. Correlation was made with post-conceptional age and perioperative clinical variables. In preterm neonates with CHD, there were age-related differences in thalamic diffusivity (axial and radial) compared to the preterm and term non-CHD group, in contrast to no differences in anisotropy. Contrary to our hypothesis, abnormal thalamic and optic radiation microstructure was most strongly associated with an elevated first arterial blood gas pO2 and elevated preoperative arterial blood gas pH (p ¡ 0.05). Age-related thalamic microstructural abnormalities were observed in preterm neonates with CHD. Perinatal hyperoxemia and increased perioperative serum pH were associated with abnormal thalamic microstructure in preterm neonates with CHD. This study emphasizes the vulnerability of thalamocortical development in the preterm neonate with CHD.

Developmental synergy between thalamic structure and interhemispheric connectivity in the visual system of preterm infants.

Ceschin R, Wisnowski JL, Paquette LB, Nelson MD, Bluml S, Panigrahy A.

Neuroimage Clin. 2015 Jun 4;8:462-72. doi: 10.1016/j.nicl.2015.05.014. eCollection 2015.

Thalamic structural co-variation with cortical regions has been demonstrated in preterm

infants, but its relationship to cortical function and severity of non-cystic white matter injury (non-cystic WMI) is unclear. The relationship between thalamic morphology and both cortical network synchronization and cortical structural connectivity has not been established. We tested the hypothesis that in preterm neonates, thalamic volume would correlate with primary cortical visual function and microstructural integrity of cortico-cortical visual association pathways. A total of 80 term-equivalent preterm and 44 term-born infants underwent high-resolution structural imaging coupled with visual functional magnetic resonance imaging or diffusion tensor imaging. There was a strong correlation between thalamic volume and primary visual cortical activation in preterms with non-cystic WMI (r = 0.81, p-value = 0.001). Thalamic volume also correlated strongly with interhemispheric cortico-cortical connectivity (splenium) in preterm neonates with a relatively higher severity of non-cystic WMI (p-value ¡ 0.001). In contrast, there was lower correlation between thalamic volume and intrahemispheric cortico-cortical connectivity, including the inferior longitudinal fasciculus and inferior frontal orbital fasciculus. This study shows distinct temporal overlap in the disruption of thalamo-cortical and interhemispheric cortico-cortical connectivity in preterm infants suggesting developmental synergy between thalamic morphology and the emergence of cortical networks in the last trimester.

Thalamic alterations in preterm neonates and their relation to ventral striatum disturbances revealed by a combined shape and pose analysis.

Lao Y, Wang Y, Shi J, **Ceschin R**, Nelson MD, Panigrahy A, Lepore N.

Brain Struct Funct. 2016 Jan;221(1):487-506. doi: 10.1007/s00429-014-0921-7. Epub 2014 Nov 1.

Finding the neuroanatomical correlates of prematurity is vital to understanding which structures are affected, and to designing efficient prevention and treatment strategies. Converging results reveal that thalamic abnormalities are important indicators of prematurity. However, little is known about the localization of the abnormalities within the subnuclei of the thalamus, or on the association of altered thalamic development with other deep gray matter disturbances. Here, we aim to investigate the effect of prematurity on the thalamus and the putamen in the neonatal brain, and further investigate the associated abnormali-

ties between these two structures. Using brain structural magnetic resonance imaging, we perform a novel combined shape and pose analysis of the thalamus and putamen between 17 preterm (41.12 5.08 weeks) and 19 term-born (45.51 5.40 weeks) neonates at term equivalent age. We also perform a set of correlation analyses between the thalamus and the putamen, based on the surface and pose results. We locate significant alterations on specific surface regions such as the anterior and ventral anterior (VA) thalamic nuclei, and significant relative pose changes of the left thalamus and the right putamen. In addition, we detect significant association between the thalamus and the putamen for both surface and pose parameters. The regions that are significantly associated include the VA, and the anterior and inferior putamen. We detect statistically significant surface deformations and pose changes on the thalamus and putamen, and for the first time, demonstrate the feasibility of using relative pose parameters as indicators for prematurity in neonates. Our methods show that regional abnormalities of the thalamus are associated with alterations of the putamen, possibly due to disturbed development of shared pre-frontal connectivity. More specifically, the significantly correlated regions in these two structures point to frontal-subcortical pathways including the dorsolateral prefrontal-subcortical circuit, the lateral orbitofrontal-subcortical circuit, the motor circuit, and the oculomotor circuit. These findings reveal new insight into potential subcortical structural covariates for poor neurodevelopmental outcomes in the preterm population.

A multivariate surface-based analysis of the putamen in premature newborns: regional differences within the ventral striatum.

Shi J, Wang Y, **Ceschin R**, An X, Lao Y, Vanderbilt D, Nelson MD, Thompson PM, Panigrahy A, Lepore N.

PLoS One. 2013 Jul 3;8(7):e66736. doi: 10.1371/journal.pone.0066736. Print 2013.

Many children born preterm exhibit frontal executive dysfunction, behavioral problems including attentional deficit/hyperactivity disorder and attention related learning disabilities. Anomalies in regional specificity of cortico-striato-thalamo-cortical circuits may underlie deficits in these disorders. Nonspecific volumetric deficits of striatal structures have been documented in these subjects, but little is known about surface deformation in these

structures. For the first time, here we found regional surface morphological differences in the preterm neonatal ventral striatum. We performed regional group comparisons of the surface anatomy of the striatum (putamen and globus pallidus) between 17 preterm and 19 term-born neonates at term-equivalent age. We reconstructed striatal surfaces from manually segmented brain magnetic resonance images and analyzed them using our in-house conformal mapping program. All surfaces were registered to a template with a new surface fluid registration method. Vertex-based statistical comparisons between the two groups were performed via four methods: univariate and multivariate tensor-based morphometry, the commonly used medial axis distance, and a combination of the last two statistics. We found statistically significant differences in regional morphology between the two groups that are consistent across statistics, but more extensive for multivariate measures. Differences were localized to the ventral aspect of the striatum. In particular, we found abnormalities in the preterm anterior/inferior putamen, which is interconnected with the medial orbital/prefrontal cortex and the midline thalamic nuclei including the medial dorsal nucleus and pulvinar. These findings support the hypothesis that the ventral striatum is vulnerable, within the cortico-stiato-thalamo-cortical neural circuitry, which may underlie the risk for long-term development of frontal executive dysfunction, attention deficit hyperactivity disorder and attention-related learning disabilities in preterm neonates.

C.4 BRAIN DYSPLASIA AND CILIARY DYSFUNCTION IN INFANTS WITH CHD

Brain Dysplasia Associated with Ciliary Dysfunction in Infants with Congenital Heart Disease.

Panigrahy A, Lee V, **Ceschin R**, Zuccoli G, Beluk N, Khalifa O, Votava-Smith JK, DeBrunner M, Munoz R, Domnina Y, Morell V, Wearden P, Sanchez De Toledo J, Devine W, Zahid M, Lo CW.

J Pediatr. 2016 Aug 26. pii: S0022-3476(16)30643-6. doi: 10.1016/j.jpeds.2016.07.041. [Epub ahead of print]

We recruited 35 infants with CHD preoperatively and performed nasal tissue biopsy to assess respiratory CM by videomicroscopy. Cranial ultrasound scan and brain magnetic resonance imaging were obtained pre- and/or postoperatively and systematically reviewed for brain abnormalities. Segmentation was used to quantitate cerebrospinal fluid and regional brain volumes. Perinatal and perioperative clinical variables were collected. A total of 10 (28.5%) patients with CHD had abnormal CM. Abnormal CM was not associated with brain injury but was correlated with increased extraaxial cerebrospinal fluid volume ($P<.001$), delayed brain maturation ($P<.05$), and a spectrum of subtle dysplasia including the hippocampus ($P <.0078$) and olfactory bulb ($P<.034$). Abnormal CM was associated with higher composite dysplasia score ($P <.001$), and both were correlated with elevated preoperative serum lactate ($P<.001$). Abnormal respiratory CM in infants with CHD is associated with a spectrum of brain dysplasia. These findings suggest that ciliary defects may play a role in brain dysplasia in patients with CHD and have the potential to prognosticate neurodevelopmental risks.

Bibliography

[1] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2323, 1998.

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Advances In Neural Information Processing Systems*, pages 1–9, 2012.

[3] Andrew Janowczyk, Ajay Basavanhally, and Anant Madabhushi. Stain Normalization using Sparse AutoEncoders (StaNoSA): Application to digital pathology. *Computerized Medical Imaging and Graphics*, 57:50–61, 2017.

[4] Jeremy Kawahara and Ghassan Hamarneh. Multi-resolution-Tract CNN with Hybrid Pretrained and Skin-Lesion Trained Layers. pages 164–171. Springer, Cham, oct 2016.

[5] Jens Kleesiek, Gregor Urban, Alexander Hubert, et al. Deep MRI brain extraction: A 3D convolutional neural network for skull stripping. *NeuroImage*, 129:460–469, apr 2016.

[6] Tom Brosch, Lisa Y.W. Tang, Youngjin Yoo, et al. Deep 3D Convolutional Encoder Networks With Shortcuts for Multiscale Feature Integration Applied to Multiple Sclerosis Lesion Segmentation. *IEEE Transactions on Medical Imaging*, 35(5):1229–1239, may 2016.

[7] Cinly Ooi, Edward T Bullmore, Alle-Meije Wink, et al. CamBAfx: Workflow Design, Implementation and Application for Neuroimaging. *Frontiers in neuroinformatics*, 3(AUG):27, 2009.

[8] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *Bernoulli*, (1341):1–13, 2009.

[9] Johanna Calderon and David C. Bellinger. Executive function deficits in congenital heart disease: why is intervention important? *Cardiology in the Young*, (JANUARY):1–9, 2015.

[10] Catherine J. Stoodley and Catherine Limperopoulos. Structurefunction relationships in

the developing cerebellum: Evidence from early-life cerebellar injury and neurodevelopmental disorders. *Seminars in Fetal and Neonatal Medicine*, pages 1–9, 2016.

[11] William T Mahle, Federica Tavani, Robert A Zimmerman, et al. An MRI study of neurological injury before and after congenital heart surgery. *Circulation*, 106(12 Suppl 1):I109–14, 2002.

[12] Samuel S H Wang, Alexander D. Kloth, and Aleksandra Badura. The Cerebellum, Sensitive Periods, and Autism. *Neuron*, 83(3):518–532, 2014.

[13] Stephen A Back and Steven P Miller. Brain injury in premature neonates: A primary cerebral dysmaturation disorder?, apr 2014.

[14] Joseph J Volpe. Brain injury in premature infants: a complex amalgam of destructive and developmental disturbances. *The Lancet. Neurology*, 8(1):110–24, jan 2009.

[15] V Xydis, A Drougia, V Giapros, M Argyropoulou, and S Andronikou. Brain growth in preterm infants is affected by the degree of growth restriction at birth. *J Matern Fetal Neonatal Med*, 26(7):673–679, 2013.

[16] Inger Bocca-tjeertes, Arend Bos, Jorien Kerstjens, et al. Symmetrical and asymmetrical growth restriction in preterm-born children. *Pediatrics*, 133:e650–6, 2014.

[17] Catherine Limperopoulos, Gevorg Chilingaryan, Nancy Sullivan, et al. Injury to the premature cerebellum: Outcome is related to remote cortical development. *Cerebral Cortex*, 24(3):728–736, 2014.

[18] Catherine Limperopoulos, Gevorg Chilingaryan, Nicolas Guizard, et al. Cerebellar injury in the premature infant is associated with impaired growth of specific cerebral regions. *Pediatric research*, 68(2):145–150, 2010.

[19] Christopher R. Pierson and Fahd Al Sufiani. Preterm birth and cerebellar neuropathology. *Seminars in Fetal and Neonatal Medicine*, pages 1–7, 2016.

[20] Emily W Y Tam. Potential mechanisms of cerebellar hypoplasia in prematurity. *Neuroradiology*, 55(SUPPL. 2):41–46, 2013.

[21] Deanne K Thompson, Cristina Omizzolo, Christopher Adamson, et al. Longitudinal growth and morphology of the hippocampus through childhood: Impact of prematurity and implications for memory and learning. *Human brain mapping*, 35(8):4129–39, aug 2014.

[22] James H. Cole, Maria Laura Filippetti, Matthew P G Allin, et al. Subregional Hippocampal Morphology and Psychiatric Outcome in Adolescents Who Were Born Very Preterm and at Term. *PloS one*, 10(6):e0130094, jan 2015.

[23] Emma G. Duerden, Ting Guo, Lorin Dodbiba, et al. Midazolam dose correlates with abnormal hippocampal growth and neurodevelopmental outcome in preterm infants. *Annals of Neurology*, 79(4):548–59, may 2016.

[24] Jan H. Hansen, Ina Rotermann, Jana Logoteta, et al. Neurodevelopmental outcome in hypoplastic left heart syndrome: Impact of perioperative cerebral tissue oxygenation of the Norwood procedure. *The Journal of Thoracic and Cardiovascular Surgery*, 151(5):1358–1366, may 2016.

[25] Anne Sarajuuri, Eero Jokinen, Leena Mildh, et al. Neurodevelopmental burden at age 5 years in patients with univentricular heart. *Pediatrics*, 130(6):e1636–46, dec 2012.

[26] Sarah Tabbutt, Alex S Nord, Gail P Jarvik, et al. Neurodevelopmental outcomes after staged palliation for hypoplastic left heart syndrome. *Pediatrics*, 121(3):476–83, mar 2008.

[27] Michael von Rhein, Andreas Buchmann, Cornelia Hagmann, et al. Severe Congenital Heart Defects Are Associated with Global Reduction ofNeonatal Brain Volumes. *The Journal of pediatrics*, 167(6):1259–1263.e1, 2015.

[28] Patrick S. McQuillen, A. James Barkovich, Shannon E G Hamrick, et al. Temporal and anatomic risk profile of brain injury with neonatal repair of congenital heart defects. *Stroke*, 38(2 PART 2):736–741, 2007.

[29] Jane W. Newburger, Lynn A. Sleeper, David C. Bellinger, et al. Early developmental outcome in children with hypoplastic left heart syndrome and related anomalies: The single ventricle reconstruction trial. *Circulation*, 125(17):2081–2091, 2012.

[30] Cynthia Ortinau, John Beca, Jennifer Lambeth, et al. Regional alterations in cerebral growth exist preoperatively in infants with congenital heart disease. *The Journal of thoracic and cardiovascular surgery*, 143(6):1264–70, jun 2012.

[31] Thomas A. Miller, Victor Zak, Peter Shrader, et al. Growth Asymmetry, Head Circumference, and Neurodevelopmental Outcomes in Infants with Single Ventricles. *Journal of Pediatrics*, 168:220–225.e1, 2015.

[32] Joseph J Volpe. Cerebellum of the premature infant: rapidly developing, vulnerable, clinically important. *Journal of child neurology*, 24(9):1085–104, sep 2009.

[33] Cynthia Ortinau, Terrie Inder, Jennifer Lambeth, et al. Congenital heart disease affects cerebral size but not brain growth. *Pediatric Cardiology*, 33(7):1138–1146, oct 2012.

[34] Audrey Putoux, Sophie Thomas, Karlien L M Coene, et al. KIF7 mutations cause fetal hydrolethalus and acrocallosal syndromes. *Nature genetics*, 43(6):601–6, jun 2011.

[35] Angeliki Louvi, ElizabethA. A. E.A. Grove, P. Aanstad, et al. Cilia in the CNS: The Quiet Organelle Claims Center Stage. *Neuron*, 69(6):1046–1060, mar 2011.

[36] Nader Nakhleh, Richard Francis, Rachel A Giese, et al. High prevalence of respiratory ciliary dysfunction in congenital heart disease patients with heterotaxy. *Circulation*, 125(18):2232–2242, may 2012.

[37] You Li, Nikolai T. Klena, George C. Gabriel, et al. Global genetic analysis in mice unveils central role for cilia in congenital heart disease. *Nature*, 521(7553):520–524, mar 2015.

[38] Calvin S Carter, Timothy W Vogel, Qihong Zhang, et al. Abnormal development of NG2+PDGFR-α+ neural progenitor cells leads to neonatal hydrocephalus in a ciliopathy mouse model. *Nature Medicine*, 18(12):1797–1804, nov 2012.

[39] Ashok Panigrahy, Vincent Lee, Rafael Ceschin, et al. Brain Dysplasia Associated with Ciliary Dysfunction in Infants with Congenital Heart Disease. *Journal of Pediatrics*, 178:141–148.e1, aug 2016.

[40] T Paus, D. L. Collins, A. C. Evans, et al. Maturation of white matter in the human brain: A review of magnetic resonance studies. *Brain Research Bulletin*, 54(3):255–266, 2001.

[41] Rhoshel K. Lenroot and Jay N. Giedd. Brain development in children and adolescents: Insights from anatomical magnetic resonance imaging. *Neuroscience and Biobehavioral Reviews*, 30(6):718–729, 2006.

[42] Gang Li, Yang Li, Yaping Wang, and Dinggang Shen. Consistent sulcal parcellation of longitudinal cortical surfaces. In *Proceedings - International Symposium on Biomedical Imaging*, volume 57, pages 337–340. NIH Public Access, jul 2011.

[43] K Kazemi and N Noorizadeh. Quantitative Comparison of SPM, FSL, and Brainsuite for Brain MR Image Segmentation. *Journal of biomedical physics & engineering*, 4(1):13–26, mar 2014.

[44] Peter Kochunov, William Rogers, Jean-Francois Francois Mangin, and Jack Lancaster. A library of cortical morphology analysis tools to study development, aging and genetics of cerebral cortex. *Neuroinformatics*, 10(1):81–96, jan 2012.

[45] Bruce Fischl, André Van Der Kouwe, Christophe Destrieux, et al. Automatically Parcellating the Human Cerebral Cortex. *Cerebral Cortex*, 14(1):11–22, jan 2004.

[46] Bruce Fischl, David H Salat, Evelina Busa, et al. Whole brain segmentation: Automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33(3):341–355, jan 2002.

[47] Ioannis S. Gousias, A. David Edwards, Mary A. Rutherford, et al. Magnetic resonance imaging of the newborn brain: Manual segmentation of labelled atlases in term-born and preterm infants. *NeuroImage*, 62(3):1499–1509, 2012.

[48] Ioannis S. Gousias, Alexander Hammers, Serena J. Counsell, et al. Magnetic Resonance Imaging of the Newborn Brain: Automatic Segmentation of Brain Images into 50 Anatomical Regions. *PLoS ONE*, 8(4):e59990, apr 2013.

[49] Michael A. Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2015.

[50] Jacob Devlin, Hao Cheng, Hao Fang, et al. Language Models for Image Captioning: The Quirks and What Works. may 2015.

[51] Pengtao Xie, Misha Bilenko, Tom Finley, et al. Crypto-Nets: Neural Networks over Encrypted Data. In *ICLR 2015*, pages 1–9, dec 2014.

[52] Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. Multi-task Sequence to Sequence Learning. *Iclr*, (c):1–9, nov 2016.

[53] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial Transformer Networks. jun 2015.

[54] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *ImageNet Challenge*, pages 1–10, sep 2014.

[55] Tom Sercu, Christian Puhrsch, Brian Kingsbury, and Yann Lecun. Very deep multilingual convolutional neural networks for LVCSR. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2016-May, pages 4955–4959, sep 2016.

[56] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, et al. A Survey on Deep Learning in Medical Image Analysis. (1995), 2017.

[57] Yoshua Bengio. Practical Recommendations for Gradient-Based Training of Deep Architectures. 2012.

[58] Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, page 19, 2016.

[59] Ashish Agarwal, Paul Barham, Eugene Brevdo, et al. TensorFlow : Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2015.

[60] Ronan Collobert, Koray Kavukcuoglu, and Clément Farabet. Torch7: A matlab-like environment for machine learning. *BigLearn, NIPS Workshop*, pages 1–6, 2011.

[61] Stephen M. Smith. Fast robust automated brain extraction. *Human Brain Mapping*, 17(3):143–155, nov 2002.

[62] Y. Zhang, M. Brady, and S. Smith. Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1):45–57, 2001.

[63] Uroš Vovk, Franjo Pernuš, and Boštjan Likar. A review of methods for correction of intensity inhomogeneity in MRI. *IEEE Transactions on Medical Imaging*, 26(3):405–421, 2007.

[64] W. M. Wells, W. E L Crimson, R. Kikinis, and F. A. Jolesz. Adaptive segmentation of mri data. *IEEE Transactions on Medical Imaging*, 15(4):429–442, 1996.

[65] Brian B. Avants, Nicholas J. Tustison, Gang Song, et al. A reproducible evaluation of ANTs similarity metric performance in brain image registration. *NeuroImage*, 54(3):2033–2044, 2011.

[66] B. B. Avants, C. L. Epstein, M. Grossman, and J. C. Gee. Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain. *Medical Image Analysis*, 12(1):26–41, 2008.

[67] Antonios Makropoulos, Ioannis S. Gousias, Christian Ledig, et al. Automatic whole brain MRI segmentation of the developing neonatal brain. *IEEE Transactions on Medical Imaging*, 33(9):1818–1831, 2014.

[68] R W Cox. AFNI: software for analysis and visualization of functional magnetic resonance neuroimages. *Comput. Biomed. Res.*, 29(3):162–173, jun 1996.

[69] Ahmed Serag, Paul Aljabar, Gareth Ball, et al. Construction of a consistent high-definition spatio-temporal atlas of the developing brain using adaptive kernel regression. *NeuroImage*, 59(3):2255–2265, 2012.

[70] Bb Avants, Nick Tustison, and Gang Song. Advanced Normalization Tools (ANTS). *Insight Journal*, pages 1–35, 2009.

[71] Brian B Avants, Nicholas J Tustison, Michael Stauffer, et al. The Insight ToolKit image registration framework. *Frontiers in neuroinformatics*, 8(April):44, 2014.

[72] Krzysztof Gorgolewski, Christopher D Burns, Cindee Madison, et al. Nipype: a flexible, lightweight and extensible neuroimaging data processing framework in python. *Front Neuroinform*, 5, 2011.

[73] Paul A Yushkevich, Joseph Piven, Heather Cody Hazlett, et al. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *NeuroImage*, 31(3):1116–1128, jul 2006.

[74] Kelly H Zou, Simon K Warfield, Aditya Bharatha, et al. Statistical Validation of Image Segmentation Quality Based on a Spatial Overlap Index. *Academic Radiology*, 11(2):178–189, feb 2004.

[75] R Development Core Team. R: A Language and Environment for Statistical Computing, 2008.

[76] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, et al. Intriguing properties of neural networks. dec 2013.

[77] A. James Barkovich and Charles Raybaud. *Pediatric Neuroimaging*. LWW, 2012.

[78] A. Wong, T. Chavez, S O 'neil, et al. Synchronous Aberrant Cerebellar and Opercular Development in Fetuses and Neonates with Congenital Heart Disease: Correlation with Early Communicative Neurodevelopmental Outcomes, Initial Experience. *Am J Perinatol Rep*, 7(01):17–27, feb 2017.

[79] Egidio D'Angelo and Stefano Casali. Seeking a unified framework for cerebellar function and dysfunction: from circuit operations to cognition. *Frontiers in neural circuits*, 6(January):116, 2012.

[80] Jeremy D. Schmahmann and Janet C. Sherman. The cerebellar cognitive affective syndrome. *Brain*, 121(4):561–579, 1998.